Fundamenta Informaticae XX (2016) 1–35
DOI 10.3233/FI-2016-0000
IOS Press

Pebble games with algebraic rules*

Anuj Dawar

Computer Laboratory
University of Cambridge
`anuj.dawar@cl.cam.ac.uk`

Bjarki Holm

Twigkit ltd.
`bjarki@twigkit.com`

Abstract. We define a general framework of *partition games* for formulating two-player pebble games over finite structures. The framework we introduce includes as special cases the pebble games for finite-variable logics with and without counting. It also includes a *matrix-equivalence game*, introduced here, which characterises equivalence in the finite-variable fragments of the matrix-rank logic of [Dawar et al. 2009]. We show that one particular such game in our framework, which we call the *invertible-map game*, yields a family of polynomial-time approximations of graph isomorphism that is strictly stronger than the well-known Weisfeiler-Leman method. We show that the equivalence defined by this game is a refinement of the equivalence defined by each of the games for finite-variable logics.

Contents

*Research supported by EPSRC grant EP/H026835/1. An extended abstract of this paper appeared in the proceedings of ICALP 2012.

1. Introduction

An important open problem in finite model theory is that of finding a logical characterisation of polynomial-time computability. That is to say, to find a logic in which a class of finite structures is expressible if, and only if, membership in the class is decidable in deterministic polynomial time (PTIME). The exact formulation of the problem (see [1]) requires additional effectivity conditions which need not concern us here. By a result proved independently by Immerman [2] and Vardi [3], it is known that inflationary fixed-point logic (IFP) expresses exactly the polynomial-time properties of *ordered* finite structures, but falls short of expressing the polynomial-time properties of *all* finite structures. It was at one time conjectured by Immerman that the extension of inflationary fixed-point logic with a mechanism for counting (IFPC) suffices for expressing all of PTIME. However, this turns out not to be the case and a counter-example was constructed by Cai, Fürer and Immerman [4]. In [5] we introduced the extension of inflationary fixed-point logic with matrix-rank operators (IFPR), after noting that the Cai-Fürer-Immerman construction, and various other examples of properties in PTIME that are not definable in IFPC, can be reduced to testing the solvability of systems of linear equations. The logic IFPR strictly extends the expressive power of IFPC while still being contained in PTIME. Grädel and Pakusa [6] have recently shown that the logic as defined in [5] is strictly contained in an alternative version of a fixed-point logic with rank operators (which they call FPR*) and therefore strictly contained in PTIME.

In this context, the study of ever more expressive logics has gone hand in hand with the development of tools for proving limitations on those logics. An important class of such tools are the so-called pebble games, which are variations and extensions of the Ehrenfeucht-Fraïssé game for first-order logic. In particular, the *k-pebble game* characterises the relation $\equiv_k^L$ of equivalence in first-order logic with k variables. Since it can be shown that any formula of IFP is invariant under $\equiv_k^L$ for some k, this becomes useful in proving inexpressibility results for IFP. Similarly, inexpressibility results for IFPC are established by showing that a property is not invariant under $\equiv_k^C$ for any k, where $\equiv_k^C$

denotes the relation of equivalence in first-order logic with counting quantifiers and at most k variables. The relation $\equiv_k^C$ has been characterised by two different pebble games: the *counting game* of Immerman and Lander [7] and the *bijection game* of Hella [8].

In addition to providing a tool for the analysis of logics, these games also provide interesting approximations of the graph isomorphism relation. In particular, it can be shown that the equivalence relation $\equiv_{k+1}^C$ is exactly the relation decided by the k-dimensional Weisfeiler-Leman method (see [4] for a description of the method and its relationship with $\equiv_k^C$). This is a family of polynomial-time algorithms which approach graph isomorphism in the limit by ever finer approximations. A key contribution of the Cai-Fürer-Immerman construction of a property in PTIME that is not definable in IFPC is to show that there is no fixed k such that the k-dimensional Weisfeiler-Leman algorithm decides graph isomorphism.

In a similar way, the logic of matrix-rank operators defined in [5] yields a family of equivalence relations $\equiv_{k,m,\Omega}^R$ which provide a stratification of graph isomorphism and which can be used to analyse definability in IFPR. Here, $\equiv_{k,m,\Omega}^R$ refers to equivalence in the logic that extends k-variable first-order logic with matrix-rank operators of arity at most m for matrices over GF_p for any p in the finite set of primes Ω, wher we write GF_p to denote the finite field with p elements. One of our main contributions in this paper is a game that characterises this logical equivalence. This game, which we call the *matrix-equivalence game*, is difficult to use and it remains a challenge to deploy it to establish a property that is not closed under $\equiv_{k,m,\Omega}^R$ for any k, m and Ω. The situation should be contrasted with the logic FPR^* of rank operators used by Grädel and Pakusa [6], where the rank of a matrix is taken over a field GF_p where p may be variable. This allows one to define properties that are not invariant under $\equiv_{k,m,\Omega}^R$ for any *finite* Ω. They are able to show, using methods that are not based on games, that this variable rank operator is not definable in IFPR.

The matrix-equivalence game and the relations $\equiv_{k,m,\Omega}^R$ that it characterises suffer from another limitation as approximations of the graph isomorphism relation. Namely, it is not clear whether $\equiv_{k,m,\Omega}^R$ can be decided in polynomial time. Indeed, the natural algorithm that is obtained from the definition of the matrix-equivalence game runs in exponential time. This leads us to consider an alternative game that we call the *invertible-map game*. This game is obtained by replacing the algebraic matrix-equivalence condition with a condition of simultaneous similarity of tuples of matrices. As a result we obtain a family of equivalence relations $\approx_{m,\Omega}^k$ which refine $\equiv_{k,m,\Omega}^R$. Even though these relations are refinements of those obtained from the matrix-equivalence game, they seem easier to decide. Using a result of Chistov et al. [9] we are able to show that each of the relations $\approx_{m,\Omega}^k$ is decidable in polynomial time. Therefore, this gives us a family of *polynomial-time* algorithms which, like the Weisfeiler-Leman method, approximates isomorphism in the limit. This family is strictly stronger than the Weisfeiler-Leman method in the sense that it can also distinguish the Cai-Fürer-Immerman graphs at some fixed level.

The games we introduce in this paper are formulated as *partition games*. They are so called because the Duplicator is required at each move to give a suitable partition of the game board. This partition has to satisfy certain algebraic conditions which vary according to the game we are considering. It turns out that the games for $\equiv_k^L$ and $\equiv_k^C$ can be formulated as partition games, by replacing the algebraic rules of the matrix-equivalence game with weaker conditions that the partitions need to satisfy. This provides a general framework for exploring other games and, indeed, other equivalence relations on structures. So far, model-comparison games have been formulated for specific logics. Perhaps we can reverse this and extract suitable logics from well-behaved games? One such challenge is to formulate a logic that corresponds to the invertible map game that we define here.

2. Preliminaries

For a positive integer k, we write $[k]$ to denote the set $\{1, \ldots, k\}$. We denote tuples $(v_1, \ldots, v_k)$ by $\vec{v}$ and their length by $\|\vec{v}\|$. If $\vec{v}$ is a k-tuple of elements from a set X, $i \in [k]$ and $w \in X$, then we write $\vec{v}\frac{w}{i}$ for the tuple obtained from $\vec{v}$ by replacing the i-th component with w; that is, $\vec{v}\frac{w}{i} = (v_1, \ldots, v_{i-1}, w, v_{i+1}, \ldots, v_k)$. If $m \leq k$, $\vec{i} = (i_1, \ldots, i_m) \in [k]^m$ is a tuple of distinct integers (an 'index pattern') and $\vec{w}$ is an m-tuple of elements from X, then we write $\vec{v}\frac{\vec{w}}{\vec{i}} := \vec{v}\frac{w_1}{i_1} \cdots \frac{w_m}{i_m}$.

Recall that a *partition* of a set A is a collection $\mathbf{P}$ of non-empty and mutually disjoint subsets of A (called *parts*) whose union is A. For $a \in A$, we write $[\![a]\!]_{\mathbf{P}}$ to denote the $\mathbf{P}$-part containing a. If $\mathbf{P}$ and $\mathbf{Q}$ are partitions of A and B, respectively, then $\mathbf{P} \times \mathbf{Q} = \{X \times Y \mid X \in \mathbf{P} \text{ and } Y \in \mathbf{Q}\}$ is a partition of $A \times B$ (the *product partition* by $\mathbf{P}$ and $\mathbf{Q}$). For two partitions $\mathbf{P}$ and $\mathbf{Q}$ of A, we say that $\mathbf{P}$ *refines* $\mathbf{Q}$, and write $\mathbf{P} \leqslant \mathbf{Q}$, if for every $P \in \mathbf{P}$ there is some $Q \in \mathbf{Q}$ with $P \subseteq Q$. In this case, we also say that $\mathbf{Q}$ is a *coarsening* of $\mathbf{P}$. Each partition $\mathbf{P}$ of A is naturally associated with a uniquely determined equivalence relation on A, which relates a and b if and only if $[\![a]\!]_{\mathbf{P}} = [\![b]\!]_{\mathbf{P}}$. Conversely, an equivalence relation $\sim$ on A is associated with the partition $A/\sim$ of A into the equivalence classes of $\sim$.

2.1. Matrices indexed by unordered sets

If F is a field and I, J are finite and non-empty sets then an $I \times J$ *matrix* over F is a function $M : I \times J \to F$. If $\|I\| = \|J\|$ then we say that M is *invertible* if there is a $J \times I$ matrix N such that the product MN is the $I \times I$ identity matrix; that is, if $MN(x, y) = 1$ if $x = y$ and $MN(x, y) = 0$ otherwise [1]. In this case we refer to N as the inverse of M, denoted by M^{-1}. A matrix whose rows and columns are indexed by the same set is said to be *square*. If M is a square $I \times I$ matrix, N is a square $J \times J$ matrix and $\|I\| = \|J\|$, then we say that M and N are *equivalent* if there is an invertible $J \times I$ matrix P and an invertible $I \times J$ matrix Q such that $PMQ = N$. We write $\mathrm{GL}_{I \times J}(F)$ for the set of all $I \times J$ invertible matrices over F. Note that this set forms a ring under matrix addition and multiplication if, and only if, $I = J$.

We say that two matrices M and N are *similar* if there is an invertible $J \times I$ matrix S such that $SMS^{-1} = N$. The transformation $M \mapsto SMS^{-1}$ is called a *similarity transformation* by the similarity matrix S.

In this paper we focus on square $\{0, 1\}$-matrices whose rows and columns are indexed by tuples of elements from some finite and non-empty base set A. More specifically, if $B \subseteq A^{2m}$ for some $m \geq 1$, then we write χ_B for the characteristic function of B, seen as a $\{0, 1\}$-matrix indexed by $A^m \times A^m$. That is, χ_B is defined by $(\vec{a}, \vec{b}) \mapsto 1$ if $(\vec{a}, \vec{b}) \in B$ and $(\vec{a}, \vec{b}) \mapsto 0$ otherwise. We refer to $\chi_B(\vec{a}, \vec{b})$ as the *characteristic matrix of* B; the underlying field and the exponent m are usually clear from the context.

For a set X, we write $\mathrm{Sym}(X)$ to denote the set of all permutations of X. For $\pi \in \mathrm{Sym}(X)$, the *permutation matrix* P_π associated with π is the $X \times X$ $\{0, 1\}$-matrix with $(P_\pi)(x, y) = 1$ just in case $\pi(y) = x$.

[1] Equivalently, it can be seen that M is invertible if there is an $J \times I$ matrix N such that NM is the $J \times J$ identity matrix.

We also consider matrices expressed as a linear combination of characteristic matrices. Let $\mathbf{P} \subseteq \wp(A^{2m})$ be a collection of subsets of A^{2m} and let $\gamma : \mathbf{P} \to F$ be a function. Then we write $M_\gamma^{\mathbf{P}}$ to denote the $A^m \times A^m$ matrix over F defined by $M_\gamma^{\mathbf{P}} := \sum_{P \in \mathbf{P}} \gamma(P) \cdot \chi_P$. Typically, $\mathbf{P}$ will be a partition of A^{2m}.

2.2. Logics and structures

A relational *vocabulary* τ is a finite sequence of relation and constant symbols $(R_1, \dots, R_k, c_1, \dots, c_\ell)$, where each R_i has a fixed *arity* $a_i \in \mathbb{N}$. A structure $\mathbf{A} = (U(\mathbf{A}), R_1^{\mathbf{A}}, \dots, R_k^{\mathbf{A}}, c_1^{\mathbf{A}}, \dots, c_\ell^{\mathbf{A}})$ over the vocabulary τ (or a τ-*structure*) consists of a non-empty set $U(\mathbf{A})$, called the *universe* of $\mathbf{A}$, together with relations $R_i^{\mathbf{A}} \subseteq U(\mathbf{A})^{a_i}$ and constants $c_j^{\mathbf{A}} \in U(\mathbf{A})$ for each $1 \leq i \leq k$ and $1 \leq j \leq \ell$. Members of the set $U(\mathbf{A})$ are called the *elements* of $\mathbf{A}$ and we define the *size* of $\mathbf{A}$ to be the cardinality of its universe. In what follows, we often consider multi-sorted structures. That is, $U(\mathbf{A})$ is given as the disjoint union of a number of different *sorts*. In this paper we consider only finite structures, that is structures over a finite universe. For a particular vocabulary τ we use $\mathrm{fin}[\tau]$ to denote the set of all finite τ-structures.

We assume that all structures are finite and that all vocabularies are finite and relational. Throughout, we commonly write τ to denote a vocabulary. We write $U(\mathbf{A})$ for the universe of a structure $\mathbf{A}$ and write $\|\mathbf{A}\|$ for the cardinality of $U(\mathbf{A})$. We denote the class of all finite τ-structures with fixed tuples of $r \in \mathbb{N}$ parameters by $\mathrm{fin}[\tau; r] := \big\{(\mathbf{A}, \vec{a}) \mid \mathbf{A} \in \mathrm{fin}[\tau],\ \vec{a} \in U(\mathbf{A})^r\big\}$. If $\alpha \subseteq \mathrm{fin}[\tau; r]$ is a class of finite τ-structures with parameters and $\mathbf{A}$ a τ-structure, then we write $\alpha_{\mathbf{A}} := \{\vec{a} \in U(\mathbf{A})^r \mid (\mathbf{A}, \vec{a}) \in \alpha\}$ to denote the r-ary relation on $\mathbf{A}$ obtained by restricting α to members involving $\mathbf{A}$.

2.3. Types and equivalences

Let L be a logic (for example, one of the fixed-point or finite-variable logics we consider below) and $\mathbf{A}$ a τ-structure. The $\mathrm{L}[\tau]$-*type* of a tuple $\vec{a} = (a_1, \dots, a_k)$ of elements of $\mathbf{A}$ is the class of all L-formulae in k free variables that are satisfied by $\vec{a}$ in $\mathbf{A}$:

$$\mathrm{tp}(\mathrm{L}; \mathbf{A}, \vec{a}) := \big\{\varphi(\vec{x}) \in \mathrm{L}[\tau] \mid \mathbf{A} \models \varphi[\vec{a}]\big\},$$

where $\vec{x}$ is a k-tuple of variables. We often use Greek symbols $\alpha, \beta, \gamma, \dots$ to denote types. We write $\mathrm{Tp}(\mathrm{L}; \tau, k)$ for the class of all $\mathrm{L}[\tau]$-types in k free variables over finite τ-structures, that is

$$\mathrm{Tp}(\mathrm{L}; \tau, k) := \big\{\mathrm{tp}(\mathrm{L}; \mathbf{A}, \vec{a}) \mid (\mathbf{A}, \vec{a}) \in \mathrm{fin}[\tau; k]\big\}.$$

Let $(\mathbf{A}, \vec{a}), (\mathbf{B}, \vec{b}) \in \mathrm{fin}[\tau; k]$, $k \geq 1$. We say $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ are L-*equivalent* if $\mathrm{tp}(\mathrm{L}; \mathbf{B}, \vec{a}) = \mathrm{tp}(\mathrm{L}; \mathbf{B}, \vec{b})$. In other words, $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ are L-equivalent if $\vec{a}$ and $\vec{b}$ satisfy exactly the same L-formulae over $\mathbf{A}$ and $\mathbf{B}$ respectively.

The *atomic type* of $\vec{a}$ over $\mathbf{A}$, $\mathrm{atp}(\mathbf{A}, \vec{a})$, is the type $\mathrm{tp}(\mathrm{L}; \mathbf{A}, \vec{a})$ when L is taken to be the quantifier-free fragment of first-order logic over the signature of $\mathbf{A}$. We write $\mathrm{atp}(\mathbf{A}, \vec{a})$ to denote the atomic type of $\vec{a}$ over $\mathbf{A}$. We say that $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ are *atomically equivalent* if $\mathrm{atp}(\mathbf{A}, \vec{a}) = \mathrm{atp}(\mathbf{B}, \vec{b})$.

The *equality type* of $\vec{a}$ over $\mathbf{A}$ is the type $\mathrm{tp}(\mathrm{L}; \mathbf{A}, \vec{a})$ when L is taken to be the quantifier-free fragment of first-order logic over the signature that consists only of the equality relation. In particular, if $\vec{a}$ and $\vec{b}$ have the same atomic type, then they also have the same equality type (over their respective structures).

If $\alpha \in \mathrm{Tp}(\mathrm{L}; \tau, k)$ and $\vec{a}$ is a k-tuple of elements from a τ-structure $\mathbf{A}$, then we say that $\vec{a}$ *realises* α *in* $\mathbf{A}$ if $\mathrm{tp}(\mathrm{L}; \mathbf{A}, \vec{a}) = \alpha$.

2.4. Fixed-point logics

In this paper, our focus is on finite-variable logics with certain generalised quantifiers, and the games that can be used to characterise such logics. From the viewpoint of finite model theory, the interest in studying finite-variable logics is chiefly to analyse the expressive power of fixed-point logic and its variants. Here we review two of the fixed-point logics that we will discuss in this paper, *fixed-point logic with counting operators* and *fixed-point logic with rank operators*.

Fixed-point logic with counting. Inflationary fixed-point logic with counting (IFPC) is an extension of inflationary fixed-point logic with the ability to express the cardinality of definable sets. The logic has two types of first-order variable: *element variables*, which range over elements of the structure on which a formula is interpreted in the usual way, and *number variables*, which range over some initial segment of the natural numbers. Below when we refer, without further qualification, to a symbol being a *variable*, we mean that either an element or a number variable may be used.

The atomic formulas of $\mathrm{IFPC}[\tau]$ are all formulas of the form: $\mu \leq \eta$, where μ, η are number variables; $s = t$ where s, t are element variables or constant symbols from τ; and $R(t_1, \ldots, t_m)$, where each t_i is either a variable or a constant symbol and R is a relation symbol of arity m. The set $\mathrm{IFPC}[\tau]$ of IFPC *formulas* over τ is built up from the atomic formulas by applying an inflationary fixed-point operator $[\mathbf{ifp}_{R,\vec{x}}\varphi](\vec{t})$; forming *counting terms* $\#_x\varphi$, where φ is a formula and x a variable; forming formulas of the kind $s = t$ and $s \leq t$ where s, t are number variables or counting terms; as well as the standard first-order operations of negation, conjunction, disjunction, universal and existential quantification. Collectively, we refer to element variables and constant symbols as *element terms*, and to number variables and counting terms as *number terms*.

For the semantics, number terms take values in $[n+1]$ and element terms take values in $U(\mathbf{A})$ where $n := \|\mathbf{A}\|$. The semantics of atomic formulas, fixed-points and first-order operations are defined as usual (c.f., e.g., [10] for details), with $\mu \leq \eta$ interpreted by the standard order on $[n+1]$. Finally, consider a counting term of the form $\#_x\varphi$, where φ is a formula and x an element variable. Here the intended semantics is that $\#_x\varphi$ denotes the number (i.e., the element of $[n+1]$) of elements that satisfy the formula φ.

In general, a formula $\varphi(\vec{x}, \vec{\mu})$ of IFPC defines a relation over $U(\mathbf{A}) \uplus [n+1]$ that is invariant under automorphisms of $\mathbf{A}$. For a more detailed definition of IFPC, we refer the reader to [10, 11].

Fixed-point logic with rank. Inflationary fixed-point logic with rank operators (IFPR) is an extension of inflationary fixed-point logic with the ability to express the rank of definable matrix relations over finite fields. Like IFPC, the logic has two types of first-order variable: element variables, which range over elements of the structure on which a formula is interpreted in the usual way, and number variables, which range over some initial segment of the natural numbers. The syntax of IFPR matches that of IFPC in every way except instead of counting terms, the logic has rules for forming *rank terms* of the form $\mathbf{rk}^p_{\vec{x},\vec{y}}(\varphi_1, \ldots, \varphi_p)$, where p is a prime number, $\varphi_1, \ldots, \varphi_p$ are formulas and $\vec{x}$, $\vec{y}$ are tuples of element variables.

The semantics of atomic formulas, inflationary fixed-points and standard first-order operations are defined just like for the corrsponding rules of IFPC. For the semantics of rank operators, consider a rank term of the form $\mathbf{rk}^p_{\vec{x},\vec{y}}(\varphi_1, \ldots, \varphi_p)$, where p is a prime number, $\varphi_1, \ldots, \varphi_p$ are formulas and $\vec{x}$, $\vec{y}$ are tuples of element variables. Suppose furthermore that $\vec{x}$ has m distinct variables and $\vec{y}$ has l distinct variables. Then the intended semantics over a τ-structure $\mathbf{A}$ is that $\mathbf{rk}^p_{\vec{x},\vec{y}}(\varphi_1, \ldots, \varphi_p)$ denotes the rank of the $U(\mathbf{A})^m \times U(\mathbf{A})^l$ matrix

$$\sum_{j=1}^{p-1} j \cdot \chi_{\varphi_j^{\mathbf{A}}} \pmod p$$

over the finite field GF_p (where the rank is an element of the number domain $[n^{\max(m,l)} + 1]$), where $n = |U(\mathbf{A})|$). For more details on IFPR, see [5, 12].

2.5. Finite-variable logics

We write L^k to denote the fragment of first-order logic using only the variables $x_1, \ldots, x_k$ and we write C^k for the extension of L^k with rules for defining counting formulae of the kind $\exists^{\geq i} x \,.\, \varphi(x)$, for $i > 0$ (for further details, see [10, 13]). For $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ in $\mathrm{fin}[\tau; r]$, we write $(\mathbf{A}, \vec{a}) \equiv^L_k (\mathbf{B}, \vec{b})$ to indicate that for any L^k-formula φ, it holds that $(\mathbf{A}, \vec{a}) \models \varphi$ if, and only if, $(\mathbf{B}, \vec{b}) \models \varphi$; the relation $\equiv^C_k$ is defined similarly for C^k.

For integers $i \geq 0, m \geq 1$ and prime p, we define a quantifier $\mathrm{rk}^{\geq i}_p$ which binds exactly $2m$ variables. If $\varphi_1, \ldots, \varphi_{p-1}$ are formulae, $\vec{x}$ and $\vec{y}$ are m-tuples of pairwise distinct variables, and $\mathbf{A}$ a structure, then we let

$$\mathbf{A} \models \mathrm{rk}^{\geq i}_p(\vec{x}, \vec{y}) \,.\, (\varphi_1, \ldots, \varphi_{p-1})$$

if, and only if, the rank of the square $U(\mathbf{A})^m \times U(\mathbf{A})^m$ matrix $\sum_{j=1}^{p-1} j \cdot \chi_{\varphi_j^{\mathbf{A}}}$ over GF_p is at least i, where we write GF_p to denote the finite field with p elements. If Ω is a finite and non-empty set of primes, then we write $R^k_{m;\Omega}$ to denote the logic built up in the same way as k-variable first-order logic, except that we have rules for constructing formulae with $2m$-ary rank quantifiers over GF_p ($p \in \Omega$) instead of the rules for first-order existential and universal quantifiers. Every formula in L^k or C^k is equivalent to one of $R^{k+1}_{\{p\};2}$ (where p is any prime), for we can simulate existential, universal and unary counting quantifiers by expressing the rank of diagonal matrices (see [5, 12, 14] for details). For instance, the formula $\exists x \,.\, \varphi(x)$ is semantically equivalent to $\mathrm{rk}^{\geq 1}_p(x, y) \,.\, (\varphi(x))$ over finite structures. We write $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ to indicate that $\mathbf{A}$ and $\mathbf{B}$ agree on all $R^k_{m;\Omega}$-formulae under the assignments $\vec{a}$ and $\vec{b}$, respectively; in other words, $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ if, and only if, $\mathrm{tp}(R^k_{m;\Omega}; \mathbf{A}, \vec{a}) = \mathrm{tp}(R^k_{m;\Omega}; \mathbf{B}, \vec{b})$. It can be shown that any formula of IFPR is *invariant* under $\equiv^R_{k,m,\Omega}$ for some k, m and Ω [5, 12, 14]. This means that for any IFPR-formula $\varphi(\vec{x})$ there are k, m and Ω, such that if $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ are structures such that $(\mathbf{A}, \vec{a}) \models \varphi$ and $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ then also $(\mathbf{B}, \vec{b}) \models \varphi$.

Remark 2.1. An alternative way to define finite-variable rank logic is to extend L^k with all rank quantifiers of arity *up to* some fixed bound, including quantifiers for non-square matrices as well as the usual first-order quantifiers $\exists$ and $\forall$. This would generally give us tighter bounds on the number of variables required to express properties using rank quantifiers. However, our interest is not so much in these finite-variable logics themselves, but rather in using them as tools for showing inexpressibility in fixed-point logic with rank operators, which means showing non-definability in $R^k_{m;\Omega}$ for *all* values of k, m and Ω. Allowing different forms of quantifiers in $R^k_{m;\Omega}$ would therefore only serve to complicate the games we develop in this paper, without providing any new insight.

2.6. Pebble games

Definability in L^k is elegantly characterised in terms of two-player games based on a game style originally developed by Ehrenfeucht and Fraïssé [15, 16]. These games were essentially given by Barwise [17] though versions were also independently presented by Immerman [2] and Poizat [18]. The game board of the k*-pebble game* consists of two structures $\mathbf{A}$ and $\mathbf{B}$ over the same vocabulary and k pebbles for each of the two structures, labelled $1, \dots, k$. The game has two players, Spoiler and Duplicator. At each round of the game, the following takes place.

1. Spoiler picks up a pebble in one of the structures (either an unused pebble or one that is already on the board) and places it on an element of the corresponding structure. For instance he[2] might take the pebble labelled by i in $\mathbf{B}$ and place it on an element of $\mathbf{B}$.

2. Duplicator must respond by placing the matching pebble in the opposite structure. In the above example, she must place the pebble labelled by i on an element of $\mathbf{A}$.

Assume at the end of the round that r pebbles have been placed and let $\{(a_i, b_i) \mid 1 \leq i \leq r\} \subseteq U(\mathbf{A}) \times U(\mathbf{B})$ denote the r pairs of pebbled elements, such that for each i the label of the pebble on element a_i is the same as the label of the pebble on element b_i. If the partial map $f : U(\mathbf{A}) \to U(\mathbf{B})$ given by

$$f := \{(a_i, b_i) \mid 1 \leq i \leq r\} \cup \{(c^{\mathbf{A}}, c^{\mathbf{B}}) \mid c \in \tau \text{ a constant}\}$$

is not a partial isomorphism, then Spoiler has won the game; otherwise it can continue for another round. We say that Duplicator has a winning strategy in the k-pebble game if she can play the game forever, maintaining a partial isomorphism at the end of each round. We also consider the situation where the game starts with some of the pebbles initially placed on the game board. Formally, we refer to a placement of pebbles over one of the structures as a *position*. If $\vec{a}$ and $\vec{b}$ are r-tuples of elements from $U(\mathbf{A})$ and $U(\mathbf{B})$ respectively, $r \leq k$, then the game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ is played as above, except that pebbles $1, \dots, r$ in $\mathbf{A}$ are initially placed on the elements $a_1, \dots, a_r$ of $\vec{a}$ and pebbles $1, \dots, r$ in $\mathbf{B}$ are initially placed on the elements $b_1, \dots, b_r$ of $\vec{b}$.

The result that links this game with definability in L^k says that Duplicator has a winning strategy in the k-pebble game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ if, and only if, $(\mathbf{A}, \vec{a}) \equiv^L_k (\mathbf{B}, \vec{b})$. This correspondence gives us a purely combinatorial method for proving inexpressibility results for k-variable logic in general and IFP in particular, since it can be shown that any formula of IFP is invariant under $\equiv^L_k$ for some k. This method can be formalised as follows:

> To show that a property P of finite structures is not definable in IFP, it suffices to show that for each $k \in \mathbb{N}$ there is a pair of structures $(\mathbf{A}_k, \mathbf{B}_k)$ for which it holds that: (*i*) $\mathbf{A}_k$ has property P but $\mathbf{B}_k$ does not; and (*ii*) Duplicator has a winning strategy in the k-pebble game on $\mathbf{A}_k$ and $\mathbf{B}_k$.

[2]By convention, Spoiler is masculine and Duplicator feminine.

Immerman and Lander [7] and Hella [8] later introduced separate versions of the k-pebble game for analysing the expressiveness of C^k over finite models. Both of these games can be used to establish lower bounds for IFPC over finite structures since it can be shown that any formula of IFPC is invariant under $\equiv_k^C$ for some k. Here we focus our attention on the game given by Hella, which we refer to as the *k-pebble bijection game*. As before, the game is played by Spoiler and Duplicator (each with k pebbles) on structures $\mathbf{A}$ and $\mathbf{B}$. If $\|\mathbf{A}\| \neq \|\mathbf{B}\|$, then Spoiler wins the game immediately. Otherwise, each round of the game proceeds as follows:

1. Spoiler picks up a pebble from $\mathbf{A}$ and the matching pebble from $\mathbf{B}$.
2. Duplicator has to respond by choosing a bijection $h : U(\mathbf{A}) \to U(\mathbf{B})$.
3. Spoiler then places the pebble chosen from $\mathbf{A}$ on some element $a \in U(\mathbf{A})$ and places the matching pebble from $\mathbf{B}$ on $h(a)$.

This completes one round in the game. If, after this round, the partial map from $\mathbf{A}$ to $\mathbf{B}$ defined by the pebbled positions (plus constants) is not a partial isomorphism, then Spoiler has won the game. Otherwise it can continue for another round. This game characterises definability in C^k in the sense that Duplicator has a winning strategy in the k-pebble bijection game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ if, and only if, $(\mathbf{A}, \vec{a}) \equiv_k^C (\mathbf{B}, \vec{b})$.

2.7. Class extensions and extension matrices

We frequently consider relations that arise by extending a fixed tuple of elements in a structure according to some criteria. For example, consider a formula φ and let $\vec{a}$ be an assignment of values to the free variables of φ over a structure $\mathbf{A}$. Then the set of all pairs (c, d) from $\mathbf{A}$ which, when used to replace the first two elements of $\vec{a}$ to give a satisfying assignment to φ, can be seen as a binary "extension" of $\vec{a}$ in $\mathbf{A}$, defined by the formula φ. Moreover, this relation can be viewed as a $\{0, 1\}$-matrix over $\mathbf{A}$ in the usual way, which gives us a way to associate a pair $(\mathbf{A}, \vec{a})$ with a family of matrices over $\mathbf{A}$.

More formally, consider a class $\alpha \subseteq \mathrm{fin}[\tau; k]$ and let $\vec{i} = (i_1, \ldots, i_n) \in [k]^n$ be a tuple of distinct integers, $n \leq k$. Then we write $\mathrm{ext}^{\alpha}_{\vec{i}}$ to denote the functor on $\mathrm{fin}[\tau; k]$ defined by $\mathrm{ext}^{\alpha}_{\vec{i}}(\mathbf{A}, \vec{a}) := \{\vec{b} \in U(\mathbf{A})^n \mid (\mathbf{A}, \vec{a}\frac{\vec{b}}{\vec{i}}) \in \alpha\}$. We refer to $\mathrm{ext}^{\alpha}_{\vec{i}}(\mathbf{A}, \vec{a})$ as the *$\vec{i}$-extension of* $(\mathbf{A}, \vec{a})$ *into* α. Abusing notation, if φ is a formula whose free variables are all amongst $\vec{x} = (x_1, \ldots, x_k)$, then we let $\mathrm{ext}^{\varphi}_{\vec{i}} := \mathrm{ext}^{\alpha_\varphi}_{\vec{i}}$, where $\alpha_\varphi := \{(\mathbf{A}, \vec{a}) \in \mathrm{fin}[\tau; k] \mid (\mathbf{A}, \vec{a}) \models \varphi\}$. That is,

$$\mathrm{ext}^{\varphi}_{\vec{i}}(\mathbf{A}, \vec{a}) = \{\vec{b} \in U(\mathbf{A})^n \mid \mathbf{A} \models \varphi[\vec{a}\tfrac{\vec{b}}{\vec{i}}]\} \subseteq U(\mathbf{A})^n.$$

If $n = 2m$, then we write $\mathrm{extmat}^{\alpha}_{\vec{i}}(\mathbf{A}, \vec{a})$ and $\mathrm{extmat}^{\varphi}_{\vec{i}}(\mathbf{A}, \vec{a})$ to denote the $U(\mathbf{A})^m \times U(\mathbf{A})^m$ characteristic matrices of $\mathrm{ext}^{\alpha}_{\vec{i}}(\mathbf{A}, \vec{a})$ and $\mathrm{ext}^{\varphi}_{\vec{i}}(\mathbf{A}, \vec{a})$, respectively. We refer to such matrices as *extension matrices*.

3. A game characterisation of rank logics

In order to analyse the expressive power of rank logics over finite structures, it is important to develop methods for proving non-definability. In this context, we consider extensions of pebble games—variations of Ehrenfeucht-Fraïssé games for first-order logic—which have assumed a central role in the study of both finite-variable and fixed-point logics. In this section we give a pebble-game characterisation of finite-variable logic with quantifiers for matrix rank. This gives us a combinatorial method for proving lower bounds (inexpressibility results) for fixed-point logic with rank operators, thereby settling one of the open problems presented in [5].

To give the intuition behind this game, we first describe a simple "partition game" that is based on the same game protocol. The partition game is played by two players, Spoiler and Duplicator, on a pair of relational structures $\mathbf{A}$ and $\mathbf{B}$, each with k pebbles labelled $1, \ldots, k$. At each round of the game, Spoiler removes a pebble from $\mathbf{A}$ and the corresponding pebble from $\mathbf{B}$. Unlike the classical pebble game, Duplicator is not allowed to move any pebbles herself. However, in response to the challenge of the Spoiler, she is allowed to divide the game board into disjoint parts in order to restrict the possible moves that Spoiler is subsequently allowed to make. More specifically, in response to Spoiler's challenge, Duplicator partitions each of $U(\mathbf{A})$ and $U(\mathbf{B})$ into the same number of disjoint parts and gives a matching between the parts in $U(\mathbf{A})$ and the parts in $U(\mathbf{B})$. Intuitively, Duplicator's strategy will be to gather in each part all those elements that lead to game positions that are sufficiently alike. In turn, Spoiler is allowed to place each of the chosen pebbles on some element of the corresponding structure, with the *restriction* that the two newly pebbled elements have to be within matching parts. That completes a round of the game. Compared with the standard Ehrenfeucht-Fraïssé-style pebble game (see [10, Ch. 3]), it may seem that the partition game is biased against the Duplicator, since she is not allowed to place her own pebbles after seeing where Spoiler places his. However, it can be shown that the two games are actually equivalent over finite structures.

The idea of dividing the game board into disjoint parts leads to a very generic template for designing new pebble games. For instance, if we adapt the rules so that any two matching parts need to have the same cardinality, then we get a game equivalent to the bijection game. The "matrix-equivalence game" we describe next is obtained by putting additional linear-algebraic constraints on the matching game parts.

3.1. Matrix-equivalence game

Let k and m be positive integers with $2m \leq k$ and let Ω be a finite and non-empty set of primes. The game board of the k-pebble m-ary *matrix-equivalence game* over Ω (or (k, m, Ω)-matrix-equivalence game for short) consists of two structures $\mathbf{A}$ and $\mathbf{B}$ of the same vocabulary, each with k pebbles labelled $1, \ldots, k$. We may start from an initial position where $r \leq k$ pebbles of $\mathbf{A}$ are already placed on the elements of an r-tuple $\vec{a}$ of elements in $\mathbf{A}$ and the corresponding r pebbles in $\mathbf{B}$ on an r-tuple $\vec{b}$ of elements in $\mathbf{B}$. If $\|\mathbf{A}\| \neq \|\mathbf{B}\|$ or the mapping defined by the initial pebble positions is not a partial isomorphism then Spoiler wins the game immediately. Otherwise, each round of the game proceeds as follows.

1. Spoiler chooses a prime $p \in \Omega$ and picks up $2m$ pebbles in some order from $\mathbf{A}$ and the $2m$ corresponding pebbles in the same order from $\mathbf{B}$.

2. Duplicator has to respond by choosing

   - a partition $\mathbf{P}$ of $U(\mathbf{A})^m \times U(\mathbf{A})^m$,
   - a partition $\mathbf{Q}$ of $U(\mathbf{B})^m \times U(\mathbf{B})^m$, with $\|\mathbf{P}\| = \|\mathbf{Q}\|$, and
   - a bijection $f : \mathbf{P} \to \mathbf{Q}$,

for which it holds that for all labellings $\gamma : \mathbf{P} \to \mathsf{GF}_p$,

$$\mathrm{rk}(M^{\mathbf{P}}_{\gamma}) = \mathrm{rk}(M^{\mathbf{Q}}_{\gamma \circ f^{-1}}). \qquad \textbf{(r.c.)}$$

Here the composite map $\gamma \circ f^{-1} : \mathbf{Q} \to \mathsf{GF}_p$ is seen as a labelling of $\mathbf{Q}$.

3. Spoiler next picks a part $P \in \mathbf{P}$ and places the $2m$ chosen pebbles from $\mathbf{A}$ on the elements of some tuple in P (in the order they were chosen earlier) and the corresponding $2m$ pebbles from $\mathbf{B}$ on the elements of some tuple in $f(P)$ (in the same order).

This completes one round in the game. If, after this exchange, the partial map from $\mathbf{A}$ to $\mathbf{B}$ defined by the pebbled positions (in addition to constants) is not a partial isomorphism, or if Duplicator is unable to produce the required partitions, then Spoiler wins the game; otherwise it can continue for another round. We say that Duplicator has a *winning strategy* in the game if she can continue playing forever, maintaining a partial isomorphism at the end of each round.

Remark 3.1. Observe that the game condition "$\mathrm{rk}(M^{\mathbf{P}}_{\gamma}) = \mathrm{rk}(M^{\mathbf{Q}}_{\gamma \circ f^{-1}})$" is the same as saying that the two matrices are *equivalent*, since rank is a complete invariant for matrix equivalence. This explains the name of the game.

Our main result here is the following theorem, which relates definability in finite-variable rank logic with a winning strategy for Duplicator in the matrix-equivalence game. A proof of the theorem is given in Section 3.2.

Theorem 3.2. Duplicator has a winning strategy in the (k, m, Ω)-matrix-equivalence game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ if, and only if, $(\mathbf{A}, \vec{a}) \equiv^{R}_{k,m,\Omega} (\mathbf{B}, \vec{b})$.

By considering initial positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ where $\vec{a}$ and $\vec{b}$ are empty tuples (that is, when all pebbles are off the board at the start of the game), we get the following result.

Corollary 3.3. Duplicator has a winning strategy in the (k, m, Ω)-matrix-equivalence game on $\mathbf{A}$ and $\mathbf{B}$ if, and only if, $\mathbf{A} \equiv^{R}_{k,m,\Omega} \mathbf{B}$.

The next lemma shows that for all m and for all finite sets of primes Ω, the equivalence on $\mathrm{fin}[\tau; k]$ given by the matrix-equivalence game refines that given by the bijection game. The proof follows from Theorem 3.2 by observing that (*i*) the relation $\equiv^{C}_{k}$ is characterised by the k-pebble bijection game [8] and (*ii*) we can simulate counting quantifiers by applying rank quantifiers to diagonal matrices (as mentioned in section 2.5). The increase in the number of pebbles needed from k to $k + 2m - 1$ is due to this simulation of counting quantifiers by rank quantifiers: to replace a counting quantifier binding a single variable we need to introduce a rank quantifier binding $2m$ variables, so in general $2m - 1$ additional variables are needed for the translation.

Lemma 3.4. If Duplicator has a winning strategy in the $(k+2m-1, m, \Omega)$-matrix-equivalence game on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ for some Ω and $m \in \mathbb{N}$, then she has a winning strategy in the k-pebble bijection game starting on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$.

From the viewpoint of finite model theory, the interest in studying the finite-variable logics $R^k_{m;\Omega}$ is mainly to analyse the expressive power of fixed-point logics with operators for matrix rank. In this context, the correspondence between $\equiv^R_{k,m,\Omega}$ and the matrix-equivalence game gives us a game-based method for proving non-definability of queries in IFPR. Compared with the game methods for IFP and IFPC that we discussed before, this proof method is however complicated by the fact that we need to consider two additional parameters—the set of primes Ω and the quantifier arity m—in addition to the number of pebbles k employed in the game:

> To show that a property P of finite structures is not definable in IFPR, it suffices to show for each $k, m \in \mathbb{N}$, with $2m \leq k$, and each finite set of primes Ω that there is a pair of structures $(\mathbf{A}_{k,m,\Omega}, \mathbf{B}_{k,m,\Omega})$ for which it holds that
>
> 1. $\mathbf{A}_{k,m,\Omega}$ has property P but $\mathbf{B}_{k,m,\Omega}$ does not; and
> 2. Duplicator has a winning strategy in the (k, m, Ω)-matrix-equivalence game on $\mathbf{A}_{k,m,\Omega}$ and $\mathbf{B}_{k,m,\Omega}$.

Finally, note that it requires much more effort to describe a winning strategy for Duplicator in the matrix-equivalence game compared with the pebble games we saw earlier. Based only on the pebbles chosen by Spoiler at the beginning of a round, Duplicator has to partition the two sides of the game board in a way that both satisfies the rank condition (**r.c.**) and which gives a satisfying response to *any* placement of pebbles by Spoiler in the subsequent move. Note in particular that once Duplicator has specified the partitions, she has no further input for the remainder of that game round.

3.2. Proof of the game characterisation

To simplify our notation, for the proof of Theorem 3.2 we consider only positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ with $\|\vec{a}\| = \|\vec{b}\| = k$; that is, positions where all the pebbles are initially placed on the board. The argument for the case when the tuples $\vec{a}$ and $\vec{b}$ have length $r < k$ is exactly the same, except that one has to distinguish at every turn between game moves made during the first k rounds and game moves in the subsequent rounds[3]. This has the effect of making the proof non-uniform, without actually providing any new insight.

In order to give the proof, we first need to establish some technical results and new notation. Consider a tuple $\Phi = (\varphi_1, \ldots, \varphi_{p-1})$ of formulae in vocabulary τ and suppose that all the formulae occurring in Φ have free variables amongst the elements of the k-tuple $\vec{x}$. Let $\vec{i} = (i_1, \ldots, i_{2m})$ be a tuple of distinct integers in $[k]$, with $2m \leq k$. Then we write $\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p$ to denote the "formula matrix" over GF_p defined by

$$\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p := \sum_{i=1}^{p-1} i \cdot \mathrm{extmat}^{\varphi_i}_{\vec{i}}(\mathbf{A}, \vec{a}) \qquad (\mathrm{mod}\ p).$$

[3]Note that it is possible to obtain a proof for $\|\vec{a}\| = \|\vec{b}\| = r \in [k-1]$ as a direct corollary of the situation when $\|\vec{a}\| = \|\vec{b}\| = k$. In this case, given r-tuples $\vec{a}$ and $\vec{b}$, one would consider the game with pebble positions $\vec{a}'$ and $\vec{b}'$, where the k-tuple $\vec{a}'$ is obtained from $\vec{a}$ by adding $k - r$ copies of a_1 at the end of the tuple (simulating the case when $k - r + 1$ pebbles are placed on element a_1) and similarly for $\vec{b}'$. Alternatively, one could consider a game board where the structures $\mathbf{A}$ and $\mathbf{B}$ are augmented with new vertices $\star_A$ and $\star_B$, respectively, disjoint from the rest of the structure. Here the idea is that a pebble placed on these special elements is to be treated as being off-the-board. This latter approach has the benefit of working for all $r \leq k$, including $r = 0$, without changing the proof in any other way. See for example Ebbinghaus and Flum [10] for an application of this idea.

That is, $\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p$ is a linear combination of the $U(\mathbf{A})^m \times U(\mathbf{A})^m$ extension matrices of the formulae φ_i, with scalar coefficients defined by the position of each formula in the tuple Φ.

In the proofs of the following few lemmas, we also frequently consider the *types* ($R^k_{m;\Omega}$ types and atomic types, in particular) realised in a particular structure. For more background on logical types, see Section 2.3.

Lemma 3.5. Suppose $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$, with $\vec{a}$ and $\vec{b}$ k-tuples of elements. Let $\vec{x}$ be a k-tuple of variables and suppose $2m \leq k$ and $p \in \Omega$. Then for all $\varphi_1, \dots, \varphi_{p-1} \in R^k_{m;\Omega}$, with $\mathrm{free}(\varphi_i) \subseteq \vec{x}$, and all tuples $\vec{i} \in [k]^{2m}$ of distinct integers, it holds that

$$\mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p) = \mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{B}, \vec{b})_p),$$

where the matrix rank is taken over GF_p and $\Phi := (\varphi_1, \dots, \varphi_{p-1})$.

Proof:
Let $(\mathbf{A}, \vec{a}) \in \mathrm{fin}[\tau; k]$. Then for all tuples $\Phi = (\varphi_1, \dots, \varphi_{p-1})$ of $R^k_{m;\Omega}$-formulae, with $\mathrm{free}(\varphi_i) \subseteq \vec{x}$, and all $(i_1, \dots, i_{2m}) \in [k]^{2m}$, the formula

$$\mathrm{rk}^{=l}_p((x_{i_1}, \dots, x_{i_m}), (x_{i_{m+1}}, \dots, x_{i_{2m}})) \,.\, (\varphi_1, \dots, \varphi_{p-1})$$

is in $\mathrm{tp}(R^k_{m;\Omega}; \mathbf{A}, \vec{a})$ exactly for the number $l := \mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p)$. The statement of the lemma now follows by considering that $\mathrm{tp}(R^k_{m;\Omega}; \mathbf{A}, \vec{a}) = \mathrm{tp}(R^k_{m;\Omega}; \mathbf{B}, \vec{b})$. ☐

We are now ready to prove Theorem 3.2. The proof is given by two separate lemmas, one for each implication.

Lemma 3.6. If $(\mathbf{A}, \vec{a}) \not\equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ then Spoiler has a winning strategy in the (k, m, Ω)-matrix-equivalence game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$.

In the proof of this lemma, we show that if $(\mathbf{A}, \vec{a}) \not\equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ then Spoiler has a strategy to force the game, in a finite number of rounds, into positions that are not partially isomorphic. Spoiler's strategy is obtained by structural induction on some formula $\varphi \in R^k_{m;\Omega}$ on which the two game positions disagree; this argument broadly resembles similar proofs for the standard pebble games (see e.g. [13]). The main difficulty of the proof is to show that if Duplicator produces partitions $\mathbf{P}$ and $\mathbf{Q}$, then Spoiler can always find a part in one of the partitions that contains both tuples that satisfy φ and tuples that satisfy $\neg\varphi$. Once he has identified such a part, Spoiler can place his pebbles in a way that ensures that the resulting game positions disagree on a formula of quantifier rank less than φ. This gives him a strategy to win the game in a finite number of moves.

Proof:
If $(\mathbf{A}, \vec{a}) \not\equiv^R_{k,m,p} (\mathbf{B}, \vec{b})$ then there is a formula $\varphi(\vec{x}) \in R^k_{m;p}$ of quantifier rank $q \in \mathbb{N}$ such that $\mathbf{A} \models \varphi[\vec{a}]$ but $\mathbf{B} \models \neg\varphi[\vec{b}]$. If $q = 0$ then the mapping $\mathbf{A} \to \mathbf{B}$ defined by the pebbled elements $\vec{a} \mapsto \vec{b}$ is not a partial isomorphism and Spoiler has won the game. For the inductive step, suppose that $q > 0$. We show that Spoiler can in one round force the game into positions $(\mathbf{A}, \vec{a}')$ and $(\mathbf{B}, \vec{b}')$ where $(\mathbf{A}, \vec{a}')$ and $(\mathbf{B}, \vec{b}')$ can be distinguished by a formula of quantifier rank $q' < q$. This gives Spoiler a strategy

to win the game in a finite number of steps, as claimed. To establish the claim, we can assume without loss of generality that φ is of the form

$$\mathrm{rk}_p^{=l}((x_{i_1}, \ldots, x_{i_m}), (x_{i_{m+1}}, \ldots, x_{i_{2m}})) \,.\, (\varphi_1, \ldots, \varphi_{p-1})$$

for some $l \geq 0$ and $p \in \Omega$. Other cases reduce to this one through the symmetry of the claim (we have an equivalence relation), by noting that existential quantifiers can be replaced with rank quantifiers, or, if φ is a Boolean combination of formulae, by replacing φ by one of its components. Set $\vec{i} = (i_1, \ldots, i_{2m})$ and $\Phi = (\varphi_1, \ldots, \varphi_{p-1})$. Then by assumption on φ,

$$\mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p) \neq \mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{B}, \vec{b})_p). \tag{$*$}$$

Spoiler now starts the round by choosing the prime p and picking up pebbles with labels $i_1, \ldots, i_{2m}$. Duplicator has to respond by choosing partitions $\mathbf{P}$, $\mathbf{Q}$ and a bijection $f : \mathbf{P} \to \mathbf{Q}$, which satisfy the requirements of the game. If Duplicator fails to properly respond to the challenge of Spoiler, then Spoiler wins the game immediately, so assume that $\mathbf{P}$, $\mathbf{Q}$ and f satisfy the rank condition (**r.c.**). Then the following claim shows that the partitions proposed by Duplicator must contain a part with tuples that disagree on at least one of the φ_i.

Claim 3.7. There is a part $P \in \mathbf{P}$ and tuples $\vec{c} \in P$ and $\vec{d} \in f(P)$ for which there is some formula φ_i in Φ such that

$$\mathbf{A} \models \varphi_i[\vec{a}\tfrac{c_1}{i_1} \cdots \tfrac{c_{2m}}{i_{2m}}] \text{ and } \mathbf{B} \models \neg\varphi_i[\vec{b}\tfrac{d_1}{i_1} \cdots \tfrac{d_{2m}}{i_{2m}}],$$

or vice versa.

Proof of claim. Suppose, towards a contradiction, that each part $P \in \mathbf{P}$ contains only tuples that all realise one or the other, φ_i or $\neg\varphi_i$, and all tuples in $f(P)$ realise the same (corresponding) formula, for each $i \in [p-1]$. Hence, the map $\iota : \mathbf{P} \to \wp([p-1])$ that associates with each $P \in \mathbf{P}$ the set of formulae in Φ that are realised by some (and hence all) tuples in P is well-defined. Note that for each $P \in \mathbf{P}$, the formulae

$$\bigwedge_{i \in \iota(P)} \varphi_i \text{ and } \bigwedge_{i \in [p-1] \setminus \iota(P)} \neg\varphi_i$$

are realised by all tuples in P. Now consider the matrix $\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p$ defined over $\mathbf{A}$. By assumption, we can find a labelling $\gamma : \mathbf{P} \to \mathsf{GF}_p$ such that

$$\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p = M_\gamma^{\mathbf{P}} \quad \text{and} \quad \mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{B}, \vec{b})_p = M_{\gamma \circ f^{-1}}^{\mathbf{Q}}.$$

For instance, γ can be defined by $\gamma(P) := \sum_{i \in \iota(P)} i \pmod p$ for each $P \in \mathbf{P}$ (as an element of GF_p). But

$$\mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{A}, \vec{a})_p) \neq \mathrm{rk}(\mathrm{fmat}_{\vec{x},\vec{i}}(\Phi, \mathbf{B}, \vec{b})_p)$$

by $(*)$, while $\mathrm{rk}(M_\gamma^{\mathbf{P}}) = \mathrm{rk}(M_{\gamma \circ f^{-1}}^{\mathbf{Q}})$ since we assumed that Duplicator's response satisfies condition (**r.c.**). Therefore, we have a contradiction. $\lhd$

Now Spoiler picks some part P that satisfies the statement of the claim. This allows him to place the chosen pebbles on elements $(c_1, \ldots, c_{2m}) \in P$ and $(d_1, \ldots, d_{2m}) \in f(P)$ such that the two structures, with the corresponding pebble placements, can be distinguished by a formula of quantifier rank $q' < q$. $\square$

In the proof of the next lemma, we show that if $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$, then Duplicator can play one round of the (k, m, Ω)-matrix-equivalence game in a way that ensures that the resulting positions will also be $\equiv^R_{k,m,\Omega}$-equivalent. This gives her a strategy to play the game indefinitely. The idea here is to let Duplicator respond to a challenge of the Spoiler with partitions $\mathbf{P}$ and $\mathbf{Q}$ that are obtained by grouping together in each partition part all the elements realising the same $R^k_{m;\Omega}$-type (with respect to the current game positions). The bijection $f : \mathbf{P} \to \mathbf{Q}$ is similarly defined by pairing together parts whose elements all realise the same $R^k_{m;\Omega}$-type. We show that if Duplicator plays in this manner, then she can ensure both that condition (**r.c.**) is met and that Spoiler is restricted to placing his pebbles in parts which do not distinguish the two structures.

Lemma 3.8. If $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ then Duplicator has a winning strategy in the (k, m, Ω)-matrix-equivalence game starting with positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$.

Proof:
Assume that $(\mathbf{A}, \vec{a}) \equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b})$ and let $\vec{x} = (x_1, \ldots, x_k)$ be a k-tuple of distinct variables. Suppose that Spoiler begins a round of the game by choosing a prime $p \in \Omega$ and picking up pebbles labelled $i_1, \ldots, i_{2m}$, in that sequence. Hereafter, let $\vec{i} = (i_1, \ldots, i_{2m})$. Then we show that Duplicator can respond with partitions that satisfy condition (**r.c.**) and which ensure that all game positions that can result will be $\equiv^R_{k,m,\Omega}$-equivalent. Firstly, for each $\alpha \in \mathrm{Tp}(R^k_{m;\Omega}; \tau, k)$ let

$$P_\alpha := \{\vec{c} \in U(\mathbf{A})^{2m} \mid \mathrm{tp}(R^k_{m;\Omega}; \mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) = \alpha\} \text{ and}$$
$$Q_\alpha := \{\vec{d} \in U(\mathbf{B})^{2m} \mid \mathrm{tp}(R^k_{m;\Omega}; \mathbf{B}, \vec{b}\tfrac{\vec{d}}{\vec{i}}) = \alpha\}.$$

That is, each P_α consists of all $2m$-tuples that, when used to replace elements of $\vec{a}$ according to the index pattern $\vec{i}$, results in a tuple whose type over $\mathbf{A}$ is α (and similarly for each Q_α). The strategy of Duplicator is now to respond with

$$\mathbf{P} := \{P_\alpha \mid \alpha \in \Gamma \text{ and } P_\alpha \neq \emptyset\} \quad \text{and} \quad \mathbf{Q} := \{Q_\alpha \mid \alpha \in \Gamma \text{ and } Q_\alpha \neq \emptyset\},$$

where we let $\Gamma := \mathrm{Tp}(R^k_{m;\Omega}; \tau, k)$. To pair the two partitions together, Duplicator gives a mapping f on $\mathbf{P}$ defined by $P_\alpha \mapsto Q_\alpha$ for all non-empty P_α. It should be clear that $\mathbf{P}$ and $\mathbf{Q}$ are partitions of $U(\mathbf{A})^m \times U(\mathbf{A})^m$ and $U(\mathbf{B})^m \times U(\mathbf{B})^m$, respectively (just observe that each tuple of elements realises only one type).

We now establish, through a series of claims, that $\mathbf{P}$, $\mathbf{Q}$ and f satisfy the requirements (**r.c.**) of the game; in particular, that f is a bijection $\mathbf{P} \to \mathbf{Q}$.

Claim 3.9. f is a bijection $\mathbf{P} \to \mathbf{Q}$.

Proof of claim. To prove this claim, it suffices to show that $P_\alpha = \emptyset \Leftrightarrow Q_\alpha = \emptyset$ for all $\alpha \in \mathrm{Tp}(R^k_{m;\Omega}; \tau, k)$. Suppose, towards a contradiction, that there is some α such that P_α is empty while Q_α is not empty (the other case is equivalent, by symmetry of the claim). By the definition of P_α, we know that

$$\mathrm{tp}(R^k_{m;\Omega}; \mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \neq \alpha$$

for all $\vec{c} \in U(\mathbf{A})^{2m}$. This means that for each such tuple $\vec{c}$, there is some formula $\psi_{\vec{c}} \in \alpha$ such that $(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \not\models \psi_{\vec{c}}$. Fix such a formula $\psi_{\vec{c}}$ for each $\vec{c}$ and let $\Psi_\alpha := \bigwedge_{\vec{c}} \psi_{\vec{c}}$; since Ψ_α is defined by conjunction over a finite set of formulae from α, it follows that $\Psi_\alpha \in \alpha$. Since $P_\alpha = \emptyset$ it follows that

$$\|\{\vec{c} \in U(\mathbf{A})^{2m} \mid (\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \models \Psi_\alpha\}\| = 0. \qquad (\dagger)$$

This condition can be formalised in $R^k_{m;\Omega}$ as the formula

$$\theta := \mathrm{rk}_p^{=0}((x_{i_1}, \ldots, x_{i_m}), (x_{i_{m+1}}, \ldots, x_{i_{2m}})) \,.\, (\Psi_\alpha),$$

for some $p \in \Omega$. This formula asserts that the number of distinct $2m$-tuples $\vec{x}$ that realise Ψ_α over $(\mathbf{A}, \vec{a})$ is nil (more directly, that the matrix defined by Ψ_α is the all-zeroes matrix). By (†), it follows that $(\mathbf{A}, \vec{a}) \models \theta$. However, we have $(\mathbf{B}, \vec{b}) \not\models \theta$ since

$$\|\{\vec{d} \in U(\mathbf{B})^{2m} \mid (\mathbf{B}, \vec{b}\tfrac{\vec{d}}{\vec{i}}) \models \Psi_\alpha\}\| > 0,$$

by the assumption $Q_\alpha \neq \emptyset$. Observing that $\theta \in R^k_{m;\Omega}$, we conclude that

$$(\mathbf{A}, \vec{a}) \not\equiv^R_{k,m,\Omega} (\mathbf{B}, \vec{b}),$$

which contradicts the assumption of the lemma. $\lhd$

Claim 3.10. For each α for which neither P_α nor Q_α are empty, there is a formula $\varphi_\alpha \in R^k_{m;\Omega}$ (depending on both $\mathbf{P}$ and $\mathbf{Q}$) which isolates P_α in $\mathbf{P}$ and Q_α in $\mathbf{Q}$. That is, for all $\vec{c} \in P_\alpha$ and $\vec{d} \in Q_\alpha$,

$$(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \models \varphi_\alpha \quad \text{and} \quad (\mathbf{B}, \vec{b}\tfrac{\vec{d}}{\vec{i}}) \models \varphi_\alpha$$

while for all $\vec{c} \notin P_\alpha$ and $\vec{d} \notin Q_\alpha$,

$$(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \not\models \varphi_\alpha \quad \text{and} \quad (\mathbf{B}, \vec{b}\tfrac{\vec{d}}{\vec{i}}) \not\models \varphi_\alpha.$$

Proof of claim. Let P_β be non-empty and let $\vec{c} \in P_\beta$. For each $\alpha \neq \beta$ for which $P_\alpha \neq \emptyset$, fix a formula $\psi_{\vec{c}} \in \alpha$ for which $(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \not\models \psi_{\vec{c}}$. Such a formula must exist, since $\vec{c}$ does not realise the type α over $(\mathbf{A}, \vec{a})$ with respect to the index pattern $\vec{i}$. Doing this for all tuples in P_β, we let

$$\Psi^{\mathbf{P}}_{\alpha,\beta} := \bigwedge_{\vec{c} \in P_\beta} \psi_{\vec{c}}.$$

By this construction, it follows that $\Psi^{\mathbf{P}}_{\alpha,\beta} \in \alpha$. Now fix a type α with $P_\alpha \neq \emptyset$ and let

$$\varphi^{\mathbf{P}}_\alpha := \bigwedge_{\alpha \neq \beta \text{ and } P_\beta \neq \emptyset} \Psi^{\mathbf{P}}_{\alpha,\beta}.$$

Observe that for all $\vec{c} \in P_\alpha$ with $P_\alpha \neq \emptyset$, we have $(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \models \varphi^{\mathbf{P}}_\alpha$ and $(\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{i}}) \not\models \varphi^{\mathbf{P}}_\beta$ for all $\beta \neq \alpha$. In other words, $\varphi^{\mathbf{P}}_\alpha$ isolates P_α in $\mathbf{P}$.

By the same construction, we obtain for each α a formula $\varphi^{\mathbf{Q}}_\alpha$ that isolates Q_α in $\mathbf{Q}$. Then it follows that the formula $\varphi_\alpha := \varphi^{\mathbf{P}}_\alpha \wedge \varphi^{\mathbf{Q}}_\alpha \in \alpha$ isolates both P_α in $\mathbf{P}$ and Q_α in $\mathbf{Q}$. $\lhd$

Claim 3.11. $\mathrm{rk}(M^{\mathbf{P}}_\gamma) = \mathrm{rk}(M^{\mathbf{Q}}_{\gamma \circ f^{-1}})$ for all $\gamma : \mathbf{P} \to \mathsf{GF}_p$.

Proof of claim. Without loss of generality, we associate GF_p with the set $[0, p-1]$ in the canonical way throughout this proof. With that assumption, let $\gamma : \mathbf{P} \to [0, p-1]$ be a labelling. From the definition of $\mathbf{P}$, it can be seen that the collection of parts labelled $c \in [0, p-1]$ by γ corresponds to a finite set of types $\Omega_c \subseteq \mathrm{Tp}(R^k_{m;\Omega}; \tau, k)$, with each type $\alpha \in \Omega_c$ isolated by a formula $\varphi_\alpha \in R^k_{m;\Omega}$, by the previous claim. That is, for each type α it holds that

$$\alpha \in \Omega_c \Leftrightarrow \gamma(P_\alpha) = \gamma(\mathrm{ext}^{\varphi_\alpha}_{\vec{i}}(\mathbf{A}, \vec{a})) = c.$$

For $c \in [0, p-1]$, let $\psi_c := \bigvee_{\alpha \in \Omega_c} \varphi_\alpha$; clearly we have $\psi_c \in R^k_{m;\Omega}$. It can now be seen that

$$\begin{aligned} M^{\mathbf{P}}_\gamma &:= \sum_{c=0}^{p-1} c \cdot \big(\sum_{\alpha \in \Omega_c} \mathrm{extmat}^{\varphi_\alpha}_{\vec{i}}(\mathbf{A}, \vec{a}) \big) \\ &= \sum_{c=1}^{p-1} c \cdot \mathrm{extmat}^{\psi_c}_{\vec{i}}(\mathbf{A}, \vec{a}) \\ &= \mathrm{fmat}_{\vec{x},\vec{i}}(\psi_1, \ldots, \psi_{p-1}, \mathbf{A}, \vec{a})_p, \end{aligned}$$

and

$$\begin{aligned} M^{\mathbf{Q}}_{\gamma \circ f^{-1}} &:= \sum_{c=1}^{p-1} c \cdot \big(\sum_{\alpha \in \Omega_c} \mathrm{extmat}^{\varphi_\alpha}_{\vec{i}}(\mathbf{B}, \vec{b}) \big) \\ &= \sum_{c=1}^{p-1} c \cdot \\ \mathrm{extmat}^{\psi_c}_{\vec{i}}(\mathbf{B}, \vec{b}) &= \mathrm{fmat}_{\vec{x},\vec{i}}(\psi_1, \ldots, \psi_{p-1}, \mathbf{B}, \vec{b})_p. \end{aligned}$$

By Lemma 3.5 we know that

$$\mathrm{rk}\big(\mathrm{fmat}_{\vec{x},\vec{i}}(\psi_1, \ldots, \psi_{p-1}, \mathbf{A}, \vec{a})_p\big) = \mathrm{rk}\big(\mathrm{fmat}_{\vec{x},\vec{i}}(\psi_1, \ldots, \psi_{p-1}, \mathbf{B}, \vec{b})_p\big).$$

Hence, $\mathrm{rk}(M^{\mathbf{P}}_\gamma) = \mathrm{rk}(M^{\mathbf{Q}}_{\gamma \circ f^{-1}})$ over GF_p, as required. $\lhd$

By the three preceding claims, it can be seen that for any part $P \in \mathbf{P}$, any choice of elements $(c_1, \ldots, c_{2m}) \in P$ and $(d_1, \ldots, d_{2m}) \in f(P)$ that Spoiler can make will result in tuples $\vec{a}\frac{\vec{c}}{\vec{i}}$ and $\vec{b}\frac{\vec{d}}{\vec{i}}$ that realise the same $R^k_{m;\Omega}$-type. This shows that Spoiler has a strategy to maintain $\equiv^R_{k,m,\Omega}$-equivalence of game positions, which completes the proof of the lemma. $\square$

4. Playing with invertible linear maps

It follows from the pebble-game characterisation of finite-variable counting logics that the equivalence $\equiv^C_k$ on finite structures is decidable in polynomial time. Essentially, this is because the number of possible moves for Duplicator at any particular stage of the game can be inductively combined into a structural invariant that completely characterises the game equivalence, and this invariant can be constructed in polynomial time (see Otto [13] for details). In light of Theorem 3.2, we may therefore ask whether the matrix-equivalence game for $R^k_{m;\Omega}$ can be used to give a similar result for equivalence in finite-variable rank logic; that is, whether we can decide $\equiv^R_{k,m,\Omega}$ on finite structures in polynomial time. Unfortunately, unlike the case with the counting game, there does not seem to be an effective way to encode complete information about a winning strategy in the matrix-equivalence game into a polynomially-computable invariant. The main problem is the game condition (**r.c.**); this requires Duplicator to show that each pair of matrices $M^{\mathbf{P}}_{\gamma}$ and $M^{\mathbf{Q}}_{\gamma\circ f^{-1}}$ is equivalent (that is, have the same rank) but the number of these matrices is *exponential* in the size of the partition.

In an attempt to avoid this exponential number of matrix combinations, we define a modification of the matrix-equivalence game which is based on invertible linear maps. In this game, Duplicator is required to specify a bijection between the partitions of the two game structures as the conjugacy action of a single invertible matrix. In that sense, the invertible-map game can be seen as the natural extension of the bijection game for counting logics, where we replace bijections with invertible maps. In [19] it had been asked whether such a game might characterise definability in finite-variable rank logic. We show that equivalence in the invertible-map game does in fact *refine* the relations $\equiv^R_{k,m,\Omega}$ while it is not known whether the converse holds. We also establish that equivalence in the invertible-map game can be decided in polynomial time, which is not known to be true for the $\equiv^R_{k,m,\Omega}$ as we discussed above. We see one application of this new game equivalence in the next section, where we define algorithms for testing graph isomorphism by playing the invertible-map game on finite graphs.

4.1. Invertible-map game

Let k and m be positive integers with $2m \le k$ and let Ω be a finite and non-empty set of primes. The game board of the k-pebble m-ary *invertible-map game* over Ω (or (k, m, Ω)-invertible-map game for short) consists of two structures $\mathbf{A}$ and $\mathbf{B}$ of the same vocabulary, each with k pebbles labelled $1, \dots, k$ (and initial placement of pebbles $\vec{a}$ over $\mathbf{A}$ and $\vec{b}$ over $\mathbf{B}$, as before). If $\|\mathbf{A}\| \neq \|\mathbf{B}\|$ or the mapping defined by the initial pebble positions is not a partial isomorphism, then Spoiler wins the game immediately. Otherwise, each round of the game is played as follows.

1. Spoiler chooses a prime $p \in \Omega$ and picks up $2m$ pebbles in some order from $\mathbf{A}$ and the $2m$ corresponding pebbles in the same order from $\mathbf{B}$.

2. Duplicator has to respond by choosing

   - a partition $\mathbf{P}$ of $U(\mathbf{A})^m \times U(\mathbf{A})^m$,
   - a partition $\mathbf{Q}$ of $U(\mathbf{B})^m \times U(\mathbf{B})^m$, with $\|\mathbf{P}\| = \|\mathbf{Q}\|$, and
   - a non-singular $U(\mathbf{B})^m \times U(\mathbf{A})^m$ matrix S over GF_p,

   for which it holds that the map $f_S : \mathbf{P} \to \mathbf{Q}$ defined by

   $$f_S : P \mapsto Q \quad \text{iff} \quad S \cdot \chi_P \cdot S^{-1} = \chi_Q \tag{$\circledast$}$$

   is *total* and *bijective*, where we view χ_P and χ_Q as $\{0, 1\}$-matrices over GF_p.

3. Spoiler next chooses a part $P \in \mathbf{P}$ and places the $2m$ chosen pebbles from $\mathbf{A}$ on the elements of some tuple in P (in the order they were chosen earlier) and the corresponding $2m$ pebbles from $\mathbf{B}$ on the elements of some tuple in $f_S(P)$ (in the same order).

This completes one round in the game. If, after this exchange, the partial map from $\mathbf{A}$ to $\mathbf{B}$ defined by the pebbled elements is not a partial isomorphism, or if Duplicator is unable to produce the necessary triple $(\mathbf{P}, \mathbf{Q}, S)$, then Spoiler has won the game; otherwise it can continue for another round.

We write $(\mathbf{A}, \vec{a}) \approx^k_{m,\Omega} (\mathbf{B}, \vec{b})$ to denote that Duplicator has a strategy to play forever in the (k, m, Ω)-invertible-map game with starting positions $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$. Clearly, in the matrix-equivalence game it is sufficient for Duplicator to demonstrate the existence of a single similarity transformation that relates all linear combinations of partition matrices, since similar matrices have the same rank. Hence, we establish that $\approx^k_{m,\Omega}$ refines $\equiv^R_{k,m,\Omega}$ for all values of k, m and Ω.

Lemma 4.1. Duplicator has a winning strategy in the (k, m, Ω)-matrix-equivalence game starting on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ if she has a winning strategy in the (k, m, Ω)-invertible-map game starting on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$.

4.2. Game positions and types

Unlike other partition games, the invertible-map game we have defined above does not correspond directly to a logic. In this section we show that the relation $\approx^k_{m,\Omega}$ induced by the game has all the properties we would expect of an equivalence relation defined for a logic. In particular, we show (Lemma 4.2) that the relation is indeed an equivalence relation and that it is obtained as a limit of approximating relations $\sim_i$ that can be defined inductively. We also show that the relation behaves well with respect to natural operations like taking sub-tuples and extending a tuple (Lemma 4.4 and Corollary 4.7). Finally, in a result akin to a normal form in logic, we show that whenever Duplicator has a winning strategy in the invertible-map game, she has one of a specific form (Lemma 4.9).

We first consider a stratification of $\approx^k_{m,\Omega}$ by the number of rounds in the game, where we fix k, m, Ω and τ throughout this section. Specifically, we let $\sim_i$ be the binary relation on $\operatorname{fin}[\tau; k]$ defined by $(\mathbf{A}, \vec{a}) \sim_i (\mathbf{B}, \vec{b})$ if Duplicator has a strategy to play for at least i rounds in the (k, m, Ω)-invertible-map game on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$. This relation can be characterised inductively as follows, where we write $\operatorname{atp}(\mathbf{A}, \vec{a})$ to denote the atomic type of $\vec{a}$ in $\mathbf{A}$.

For the proof of the lemma below, recall that if $\alpha \subseteq \operatorname{fin}[\tau; k]$ is a class and $\vec{j} = (j_1, \ldots, j_{2m}) \in [k]^{2m}$ a tuple of distinct integers, then the $\vec{j}$-extension of $(\mathbf{A}, \vec{a})$ into α is defined by $\operatorname{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, \vec{a}) := \{\vec{c} \in U(\mathbf{A})^n \mid (\mathbf{A}, \vec{a}\frac{\vec{c}}{\vec{j}}) \in \alpha\}$. Also recall the definition of the extension matrix functor $\operatorname{extmat}^{\alpha}_{\vec{j}}$, see Section 2.7.

Lemma 4.2. For all $(\mathbf{A}, \vec{a}), (\mathbf{B}, \vec{b}) \in \operatorname{fin}[\tau; k]$ we have

(1) $(\mathbf{A}, \vec{a}) \sim_0 (\mathbf{B}, \vec{b})$ iff $\operatorname{atp}(\mathbf{A}, \vec{a}) = \operatorname{atp}(\mathbf{B}, \vec{b})$ and $|U(\mathbf{A})| = |U(\mathbf{B})|$

$(\mathbf{A}, \vec{a}) \sim_{i+1} (\mathbf{B}, \vec{b})$ iff $\underbrace{\begin{array}{l}(\mathbf{A}, \vec{a}) \sim_i (\mathbf{B}, \vec{b}) \text{ and for all } p \in \Omega \text{ and all } \vec{j} \in [k]^{2m} \\ \text{with distinct values there is an invertible matrix} \\ S \text{ over } \mathsf{GF}_p \text{ such that for all } \alpha \in \operatorname{fin}[\tau; k]/\sim_i\text{:} \\ S \cdot \operatorname{extmat}^{\alpha}_{\vec{j}}(\mathbf{A}, \vec{a}) \cdot S^{-1} = \operatorname{extmat}^{\alpha}_{\vec{j}}(\mathbf{B}, \vec{b}).\end{array}}_{(\star\star)}$

for each $i \in \mathbb{N}_0$.

(2) $\sim_i$ is an equivalence relation on $\mathrm{fin}[\tau;k]$ for each $i \in \mathbb{N}_0$.

Proof:
We prove parts (1) and (2) by simultaneous induction. Recall that if $(\mathbf{A},\vec{a})$ and $(\mathbf{B},\vec{b})$ are the current game positions, then the game may continue for another round only if $\mathrm{atp}(\mathbf{A},\vec{a}) = \mathrm{atp}(\mathbf{B},\vec{b})$; that is to say, only if the mapping defined by these positions is a partial isomorphism. This shows that the base case of the inductive definition of (1) captures the relation $\sim_0$. Clearly $\sim_0$ is an equivalence relation which proves the base case of (2). For the inductive step, suppose parts (1) and (2) hold for some i. For the proof of part (1), we consider each implication separately.

Suppose first that condition ($\star\star$) on the right-hand side of (1) holds for some positions $(\mathbf{A},\vec{a})$ and $(\mathbf{B},\vec{b})$ in the game. Suppose that at the beginning of the game round, Spoiler chooses a prime p and picks up the pebbles indexed by $j_1,\dots,j_{2m} \in [k]$ from each of the two structures. Write $\vec{j} = (j_1,\dots,j_{2m})$. By assumption, $\sim_i$ is an equivalence relation on $\mathrm{fin}[\tau;k]$. Duplicator's response is to partition $U(\mathbf{A})^m \times U(\mathbf{A})^m$ and $U(\mathbf{B})^m \times U(\mathbf{B})^m$ according to which equivalence classes of $\sim_i$ are realised when pebbles $j_1,\dots,j_{2m}$ are placed on new elements. More specifically, Duplicator responds with partitions

$$\mathbf{P} := \{\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \mid \alpha \in \mathrm{fin}[\tau;k]/\sim_i \text{ and } \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \neq \emptyset\}, \text{ and}$$
$$\mathbf{Q} := \{\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b}) \mid \alpha \in \mathrm{fin}[\tau;k]/\sim_i \text{ and } \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b}) \neq \emptyset\}.$$

By ($\star\star$), there is an invertible linear map S over GF_p for which the induced map $f_S : \mathbf{P} \to \mathbf{Q}$ is a bijection that maps each non-empty set $\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a})$ to the non-empty set $\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b})$. Thus, Duplicator can give a valid response $(\mathbf{P},\mathbf{Q},S)$ to the challenge of the Spoiler, and no matter how Spoiler subsequently places the chosen pebbles over the two structures, the resulting positions will alway be $\sim_i$-equivalent by definition of $\mathrm{ext}^{\alpha}_{\vec{j}}$.

For the other direction, assume that $(\mathbf{A},\vec{a}) \sim_{i+1} (\mathbf{B},\vec{b})$. Clearly also $(\mathbf{A},\vec{a}) \sim_i (\mathbf{B},\vec{b})$, by definition. Assume that at the beginning of the game round, Spoiler chooses a prime p and picks up the pebbles indexed by $j_1,\dots,j_{2m} \in [k]$ from each of the two structures (where we write $\vec{j} = (j_1,\dots,j_{2m})$). By assumption, Duplicator can respond with a triple $(\mathbf{P},\mathbf{Q},S)$ that allows her to continue the game for at least another i rounds. This implies that for each $\alpha \in \mathrm{fin}[\tau;k]/\sim_i$, the set $\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a})$ is a union of parts from $\mathbf{P}$. To see this, suppose instead that there is a part $P \in \mathbf{P}$ and tuples $\vec{a}_1,\vec{a}_2 \in P$ with distinct α_1,α_2 such that $\vec{a}_1 \in \mathrm{ext}^{\alpha_1}_{\vec{j}}(\mathbf{A},\vec{a})$ and $\vec{a}_2 \in \mathrm{ext}^{\alpha_2}_{\vec{j}}(\mathbf{A},\vec{a})$. Let $\vec{b}'$ be any tuple in $f_S(P)$. If $\vec{b}' \in \mathrm{ext}^{\alpha_1}_{\vec{j}}(\mathbf{B},\vec{b})$, then Spolier can play the pair $\vec{a}_2$ and $\vec{b}'$ to obtain a game position that is not in $\sim_i$ and if $\vec{b}' \in \mathrm{ext}^{\alpha_2}_{\vec{j}}(\mathbf{B},\vec{b})$, he can play the pair $\vec{a}_2$ and $\vec{b}'$ to the same effect. If $\vec{b}'$ is in neither of the two sets, either move is good for Spoiler. In each case, Spoiler arrives in one move at a position not in $\sim_i$, contraticting the assumption that the choice of $(\mathbf{P},\mathbf{Q},S)$ allows Duplicator to continue the game for at least another i rounds. A similar argument shows that and $\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b})$ is a union of parts from $\mathbf{Q}$. The statement ($\star\star$) now follows from the linearity of S.

For the proof of part (2), we assume $\sim_i$ is an equivalence relation for some i and we wish to show that $\sim_{i+1}$ is also an equivalence relation. Above we already established the equality of part (1) for $\sim_{i+1}$, from which it follows that $\sim_{i+1}$ is both reflexive and symmetric. To show transitivity, suppose that $(\mathbf{A},\vec{a}) \sim_{i+1} (\mathbf{B},\vec{b})$ and $(\mathbf{B},\vec{b}) \sim_{i+1} (\mathbf{C},\vec{c})$. Consider some $\vec{j} \in [k]^{2m}$ and let S and T be nonsingular $U(\mathbf{B})^m \times U(\mathbf{A})^m$ and $U(\mathbf{C})^m \times U(\mathbf{B})^m$ matrices, respectively, witnessing the similarity condition $(\star\star)$. That is, for all $\alpha \in \mathrm{fin}[\tau;k]/\sim_i$, it holds that

$$S \cdot \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \cdot S^{-1} = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b}) \text{ and}$$
$$T \cdot \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{B},\vec{b}) \cdot T^{-1} = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{C},\vec{c}).$$

But then the $U(\mathbf{C})^m \times U(\mathbf{A})^m$ matrix $T \cdot S$ is nonsingular and satisfies

$$(T \cdot S) \cdot \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \cdot (T \cdot S)^{-1} = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{C},\vec{c})$$

for all $\alpha \in \mathrm{fin}[\tau;k]/\sim_i$. $\square$

In particular, since $\approx^k_{m,\Omega}$ coincides with the intersection of the $\sim_i$ over all i, we obtain the following result.

Corollary 4.3. $\approx^k_{m,\Omega}$ is an equivalence relation on $\mathrm{fin}[\tau;k]$.

We note that $\approx^k_{m,\Omega}$-equivalence is preserved when considering sub-tuples of $\vec{a}$ and $\vec{b}$, as stated more formally by the following lemma.

Lemma 4.4. Suppose $(\mathbf{A},\vec{a}) \approx^k_{m,\Omega} (\mathbf{B},\vec{b})$ and let $f : [k] \to [k]$ be a function. Then $(\mathbf{A}, f(\vec{a})) \approx^k_{m,\Omega} (\mathbf{B}, f(\vec{b}))$ where we write $f(\vec{x}) := (x_{f(1)}, \ldots, x_{f(k)})$ for any k-tuple $\vec{x}$.

Proof:
Let $f : [k] \to [k]$ be a function. We show by induction on $i = 0, 1, \ldots$ that $(\mathbf{A}, f(\vec{a})) \sim_i (\mathbf{B}, f(\vec{b}))$ whenever $(\mathbf{A},\vec{a}) \sim_i (\mathbf{B},\vec{b})$, which proves the lemma.

To see this, we first observe that $f(\vec{a})$ and $f(\vec{b})$ must be atomically equivalent if $\vec{a}$ and $\vec{b}$ are atomically equivalent. Inductively, assume there is some $i \geqslant 0$ for which the claim holds and suppose $\vec{a}$ and $\vec{b}$ are tuples for which $(\mathbf{A},\vec{a}) \sim_{i+1} (\mathbf{B},\vec{b})$. Considering an arbitrary $p \in \Omega$ and $\vec{j} \in [k]^{2m}$, we want to show that condition $(\star\star)$ of Lemma 4.2 is met for the tuples $f(\vec{a})$ and $f(\vec{b})$.

Claim 4.5. Let $f^\star$ be the function $f^\star : x \mapsto x$ if $x \in \vec{j}$ and $f^\star : x \mapsto f(x)$ otherwise. Then for each $\alpha \in \mathrm{fin}[\tau;k]/\sim_i$, it holds that

$$\mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})) = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A}, f^\star(\vec{a})) \text{ and}$$
$$\mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{B}, f(\vec{b})) = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{B}, f^\star(\vec{b})).$$

Proof of claim. This follows from the fact that the extension matrices are defined by considering the result of replacing elements at the indices given by $\vec{j}$. So any action of f on those indices is effectively ignored. $\lhd$

Claim 4.6. For each $\alpha \in \text{fin}[\tau; k]/\sim_i$ and $\vec{j}$ for which $\text{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a}))$ is non-empty, there is a finite set $\Gamma \subseteq \text{fin}[\tau; k]/\sim_i$ such that

$$\begin{aligned}\text{extmat}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})) &= \sum_{\gamma \in \Gamma} \text{extmat}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a}) \text{ and} \\ \text{extmat}^{\alpha}_{\vec{j}}(\mathbf{B}, f(\vec{b})) &= \sum_{\gamma \in \Gamma} \text{extmat}^{\gamma}_{\vec{j}}(\mathbf{B}, \vec{b}).\end{aligned} \tag{$\ddagger$}$$

Proof of claim. Assume that α can be realised over $(\mathbf{A}, f(\vec{a}))$. Hence, there is at least one $2m$-tuple $\vec{c}$ for which $(\mathbf{A}, f(\vec{a})\frac{\vec{c}}{\vec{j}}) \in \alpha$. By Claim 1, we may assume that f acts as identity on all indices that occur in $\vec{j}$, so $f(\vec{a})\frac{\vec{c}}{\vec{j}} = f(\vec{a}\frac{\vec{c}}{\vec{j}})$. Let

$$\Gamma := \{\gamma \in \text{fin}[\tau; k]/\sim_i \mid (\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{j}}) \in \gamma \text{ where } \vec{c} \in A^{2m} \,.\, (\mathbf{A}, f(\vec{a})\tfrac{\vec{c}}{\vec{j}}) \in \alpha\}$$

be the finite collection of all $\sim_i$-equivalence classes obtained in this way. By the above, Γ is non-empty. Consider some $\gamma \in \Gamma$ realised by $\vec{c}$; that is, $(\mathbf{A}, \vec{a}\frac{\vec{c}}{\vec{j}}) \in \gamma$. Then we see that

$$\begin{aligned}\text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a}) &= \{\vec{x} \in U(\mathbf{A})^{2m} \mid (\mathbf{A}, \vec{a}\tfrac{\vec{x}}{\vec{j}}) \in \gamma\} \\ &= \{\vec{x} \in U(\mathbf{A})^{2m} \mid (\mathbf{A}, \vec{a}\tfrac{\vec{x}}{\vec{j}}) \sim_i (\mathbf{A}, \vec{a}\tfrac{\vec{c}}{\vec{j}})\} \\ &\subseteq \{\vec{x} \in U(\mathbf{A})^{2m} \mid (\mathbf{A}, f(\vec{a}\tfrac{\vec{x}}{\vec{j}})) \sim_i (\mathbf{A}, f(\vec{a}\tfrac{\vec{c}}{\vec{j}}))\} && (a) \\ &= \{\vec{x} \in U(\mathbf{A})^{2m} \mid (\mathbf{A}, f(\vec{a})\tfrac{\vec{x}}{\vec{j}}) \sim_i (\mathbf{A}, f(\vec{a})\tfrac{\vec{c}}{\vec{j}})\} && (b) \\ &= \text{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})),\end{aligned}$$

where we use the induction hypothesis in (*a*) and the result of Claim 1 in (*b*). We get the same result for $\mathbf{B}$ by choosing some $\vec{d}$ for which $(\mathbf{B}, \vec{b}\frac{\vec{d}}{\vec{j}}) \in \gamma$. Such a $\vec{d}$ must exist for otherwise $\text{ext}^{\gamma}_{\vec{j}}(\mathbf{B}, \vec{b}))$ is empty whilst $\text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a}))$ is not, which is not possible since $(\mathbf{A}, \vec{a}) \sim_{i+1} (\mathbf{B}, \vec{b})$ (here we are using condition ($\star\star$) of Lemma 4.2). Thus,

$$\begin{aligned}\text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a}) &\subseteq \text{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})) \text{ and} \\ \text{ext}^{\gamma}_{\vec{j}}(\mathbf{B}, \vec{b}) &\subseteq \text{ext}^{\alpha}_{\vec{j}}(\mathbf{B}, f(\vec{b}))\end{aligned}$$

for all $\gamma \in \Gamma$. Since the sets $\text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a})$ are pairwise disjoint it follows that

$$\begin{aligned}\bigcup_{\gamma \in \Gamma} \text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a}) &\subseteq \text{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})) \text{ and} \\ \bigcup_{\gamma \in \Gamma} \text{ext}^{\gamma}_{\vec{j}}(\mathbf{B}, \vec{b}) &\subseteq \text{ext}^{\alpha}_{\vec{j}}(\mathbf{B}, f(\vec{b})).\end{aligned}$$

Finally, observe that every $\vec{c} \in \text{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a}))$ occurs in $\text{ext}^{\gamma}_{\vec{j}}(\mathbf{A}, \vec{a})$ for some $\gamma \in \Gamma$; this γ is just the equivalence class of $(\mathbf{A}, \vec{a}\frac{\vec{c}}{\vec{j}})$. Thus,

$$\bigcup_{\gamma\in\Gamma} \mathrm{ext}^{\gamma}_{\vec{j}}(\mathbf{A},\vec{a}) = \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, f(\vec{a})) \text{ and}$$
$$\bigcup_{\gamma\in\Gamma} \mathrm{ext}^{\gamma}_{\vec{j}}(\mathbf{B},\vec{b}) = \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B}, f(\vec{b})).$$

The statement of the claim now follows by considering that each $\mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a})$ is just the characteristic matrix of $\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a})$, for each $\mathbf{A}$ and tuple $\vec{a}$.

$\lhd$

The lemma now follows from this last claim and the characterisation ($\star\star$) of Lemma 4.2. That is to say, since $(\mathbf{A},\vec{a}) \sim_{i+1} (\mathbf{B},\vec{b})$, we know that there is an invertible S that witnesses condition ($\star\star$) for $(\mathbf{A},\vec{a}) \sim_{i+1} (\mathbf{B},\vec{b})$. By Claim 2 and linearity, the same matrix S witnesses condition ($\star\star$) for $(\mathbf{A}, f(\vec{a})) \sim_{i+1} (\mathbf{B}, f(\vec{b}))$. $\square$

We note one important consequence of this lemma which we will refer to later on.

Corollary 4.7. Suppose $(\mathbf{A},\vec{a}) \approx^{k}_{m,\Omega} (\mathbf{B},\vec{b})$. Then
$(\mathbf{A},(a_1,\ldots,a_m,a_1,\ldots,a_m)) \approx^{k}_{m,\Omega} (\mathbf{B},(b_1,\ldots,b_m,b_1,\ldots,b_m))$.

In the following, it will be useful to formalise the kinds of response that Duplicator can make in her winning strategy in the game. Let A and B be finite and non-empty sets and let $m \geq 1$. Suppose $\mathbf{P}$ and $\mathbf{Q}$ are partitions of $A^m \times A^m$ and $B^m \times B^m$, respectively, with $\|\mathbf{P}\| = \|\mathbf{Q}\|$. If S is an invertible $B^m \times A^m$ matrix over GF_p, then we write $f_S : \mathbf{P} \to \mathbf{Q}$ for the partial function induced by S which is defined by condition ($\circledast$) of the invertible-map game; that is to say, f_S is the partial function defined by $P \mapsto Q \quad \text{iff} \quad S \cdot \chi_P \cdot S^{-1} = \chi_Q$. Finally, we let

$$\mathcal{S}_p(\mathbf{P},\mathbf{Q}) := \{S \in \mathrm{GL}_{B^m\times A^m}(\mathsf{GF}_p) \mid f_S : \mathbf{P} \to \mathbf{Q} \text{ is total and bijective }\}.$$

In other words, a valid move for Duplicator in the invertible-map game is to respond to a challenge of the Spoiler with a triple $(\mathbf{P},\mathbf{Q},S)$ where $S \in \mathcal{S}_p(\mathbf{P},\mathbf{Q})$, for the chosen prime p.

Definition 4.8. (Standard game partitions)
For a structure $\mathbf{A}$, a tuple $\vec{a} \in U(\mathbf{A})^{2m}$ and $\vec{j} \subseteq [k]^{2m}$, we write

$$\mathrm{stdpart}_{\vec{j}}(\mathbf{A},\vec{a}) := \{\mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \mid \alpha \in \mathrm{fin}[\tau;k]/\approx^{k}_{m,\Omega} \text{ and } \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A},\vec{a}) \neq \emptyset\}$$

to denote the *standard game partition* on $\mathbf{A}$ with respect to $\vec{a}$ and $\vec{j}$.

The proof of the following result follows directly from Lemma 4.2.

Lemma 4.9. (Standard response in a winning strategy)
Suppose Duplicator has a winning strategy in the (k,m,Ω)-invertible-map game on $(\mathbf{A},\vec{a})$ and $(\mathbf{B},\vec{b})$. Then she has a winning strategy whereby to any challenge $p \in \Omega$ and $\vec{j} \in [k]^{2m}$ that Spoiler can make in the next round of the game, Duplicator may respond by giving partitions

- $\mathbf{P} = \mathrm{stdpart}_{\vec{j}}(\mathbf{A},\vec{a})$,

- $\mathbf{Q} = \mathrm{stdpart}_{\vec{j}}(\mathbf{B}, \vec{b})$ and
- $S \in \mathcal{S}_p(\mathbf{P}, \mathbf{Q}) \neq \emptyset$

for which $f_S : \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{A}, \vec{a}) \mapsto \mathrm{ext}^{\alpha}_{\vec{j}}(\mathbf{B}, \vec{b})$ for all $\alpha \in \mathrm{fin}[\tau; k]/ \approx^{k}_{m,\Omega}$ where f_S is defined.

Our next aim is to establish some basic properties of standard game partitions. First, suppose that $\mathbf{P}$ is any partition of a set A^{2m} with an associated equivalence relation $\sim$. Then we say that $\mathbf{P}$ *preserves atomic types* if whenever $\vec{x} \sim \vec{y}$, it holds that $\vec{x}$ and $\vec{y}$ have the same atomic type (in other words, if each part of $\mathbf{P}$ contains only tuples of the same atomic type). In particular, this shows that $\mathbf{P}$ *preserves equality types*, which is to say that $\mathbf{P}$ refines the partition of A^{2m} according to the equality types of the $2m$-tuples. For partitions of this kind, we frequently consider the associated partition of A^m which is obtained by considering how $\mathbf{P}$ partitions the diagonal of A^{2m}. This is explained in more detail next.

Definition 4.10. (Diagonal partitions)
Let $\sim$ be an equivalence relation on $A^m \times A^m$. Then we write $\sim_m$ to denote the projection of $\sim$ onto A^m, which is the equivalence relation on A^m defined by

$$\vec{x} \sim_m \vec{y} \iff (\vec{x}, \vec{x}) \sim (\vec{y}, \vec{y}).$$

If $\mathbf{P}$ is a partition of $A^m \times A^m$ corresponding to an equivalence relation $\sim$, then we write $\mathrm{diag}(\mathbf{P}) := A^m / \sim_m$ for the partition of A^m induced by $\sim_m$.

The idea behind diagonal partitions is illustrated in Figure 1, below.

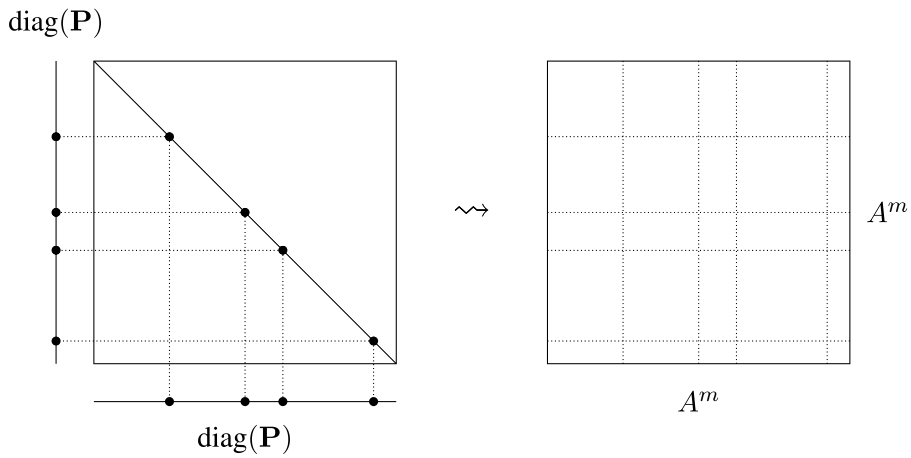

Figure 1: Consider a set A and $m \geq 1$ and suppose we have a partition $\mathbf{P}$ that preserves equality types on $A^m \times A^m$. Assume we have enumerated the elements of A^m so that each part of $\mathbf{P}$ on the diagonal $D_A = \{(\vec{a}, \vec{a}) \mid \vec{a} \in A^m\}$ appears as a continuous line segment, and indicate the diagonal parts of $\mathbf{P}$ by dots on the diagonal in the figure on the left. By taking the projection of these parts onto the row set A^m and the column set A^m, we get the partitions $\mathrm{diag}(\mathbf{P})$ of the rows and the columns, as indicated with dotted lines on the left. By considering the product partition $\mathrm{diag}(\mathbf{P}) \times \mathrm{diag}(\mathbf{P})$, we get the grid partition of $A^m \times A^m$ which is shown on the right.

Lemma 4.11. (Properties of standard game partitions)
Suppose $(\mathbf{A}, \vec{a}) \approx^k_{m,\Omega} (\mathbf{B}, \vec{b})$ and $\vec{j} \subseteq [k]^{2m}$. Let $\mathbf{P} = \mathrm{stdpart}_{\vec{j}}(\mathbf{A}, \vec{a})$ and $\mathbf{Q} = \mathrm{stdpart}_{\vec{j}}(\mathbf{B}, \vec{b})$, where $\vec{j} \subseteq [k]^{2m}$. Then both $\mathbf{P}$ and $\mathbf{Q}$ preserve atomic types and refine the grid partitions diag$(\mathbf{P}) \times$ diag$(\mathbf{P})$ and diag$(\mathbf{Q}) \times$ diag$(\mathbf{Q})$, respectively.

Proof:
The first statement follows from the fact that the standard game partitions $\mathbf{P}$ and $\mathbf{Q}$ reflect positions in the invertible-map game from where Duplicator has a strategy to play forever. The second part follows directly from Corollary 4.7. $\square$

4.3. Analysis of winning strategies

In this section we look more closely at the type of response that can be given by Duplicator according to her winning strategy in the invertible-map game. More specifically, we consider some basic structural properties of the partitions $\mathbf{P}$ and $\mathbf{Q}$ and the invertible maps S that Duplicator produces during the game play.

Our first result states that in order to win the game, Duplicator can always play by offering invertible matrices that can be part-decomposed into a direct sum of smaller matrices, according to an induced partition of the rows and columns. To explain this further, suppose at some point in a game in which Duplicator has a winning strategy, she responds to a challenge of the Spoiler by offering a triple $(\mathbf{P}, \mathbf{Q}, S)$. By Lemma 4.11, we can assume that $\mathbf{P} \leqslant \mathrm{diag}(\mathbf{P}) \times \mathrm{diag}(\mathbf{P})$ and $\mathbf{Q} \leqslant \mathrm{diag}(\mathbf{Q}) \times \mathrm{diag}(\mathbf{Q})$. Fix an enumeration $\alpha_1, \ldots, \alpha_N$ of the parts in diag$(\mathbf{P})$ and for each α_i, let β_i be the corresponding part in diag$(\mathbf{Q})$ (corresponding in the sense that the induced map f_S associates β_i with α_i). Then by the next lemma, we know that if Duplicator has a winning strategy in the invertible-map game, then she can always play by giving invertible matrices S of the following kind

$$S = \bigoplus_{i=1}^{N} S_i = \begin{array}{c c} & \begin{array}{cccc} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{array} \\ \begin{array}{c} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{array} & \begin{pmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_N \end{pmatrix} \end{array},$$

where each S_i is an invertible and *doubly stochastic*[4] $\beta_i \times \alpha_i$ matrix. This implies, in particular, that we have $S^{-1} = \bigoplus_{i=1}^{N} S_i^{-1}$.

Lemma 4.12. (Invertible matrices in a winning strategy)
Suppose $(\mathbf{A}, \vec{a}) \approx^k_{m,\Omega} (\mathbf{B}, \vec{b})$. Then Duplicator has a winning strategy that allows her to respond to any move by the Spoiler in the invertible-map game starting on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ with a triple $(\mathbf{P}, \mathbf{Q}, S)$ where

- $\mathbf{P} \leqslant \mathrm{diag}(\mathbf{P}) \times \mathrm{diag}(\mathbf{P})$ and $\mathrm{diag}(\mathbf{P}) = \{\alpha_i \mid i \in I\}$, where I is a non-empty set;
- $\mathbf{Q} \leqslant \mathrm{diag}(\mathbf{Q}) \times \mathrm{diag}(\mathbf{Q})$ and $\mathrm{diag}(\mathbf{Q}) = \{\beta_i \mid i \in I\}$ where $(\mathbf{A}, \alpha_i) \approx^k_{m,\Omega} (\mathbf{B}, \beta_i)$ for each $i \in I$; and

[4]An $I \times J$ matrix A over a commutative ring R is said to be *doubly stochastic* if every row and every column sums up to $1 \in R$; that is, if $\sum_{j \in J} A(r, j) = \sum_{i \in I} A(i, c) = 1$ for all $r \in I$ and $c \in J$.

- $S = \bigoplus_{i \in I} S_i$ is a block diagonal matrix with each S_i a matrix indexed by $\beta_i \times \alpha_i$ which is non-singular and doubly stochastic.

Remark 4.13. It follows from this lemma that the combined matrix S that is played by Duplicator, and not just its direct summands, is doubly stochastic over the chosen prime field.

Proof:
Assuming $(\mathbf{A}, \vec{a}) \approx^k_{m,\Omega} (\mathbf{B}, \vec{b})$, suppose that, as a part of her winning strategy, Duplicator responds to Spoiler choosing a prime $p \in \Omega$ and pebble positions $\vec{j}$ by giving a triple $(\mathbf{P}, \mathbf{Q}, S)$. By Lemma 4.9, we may assume that $\mathbf{P} = \mathrm{stdpart}_{\vec{j}}(\mathbf{A}, \vec{a})$ and $\mathbf{Q} = \mathrm{stdpart}_{\vec{j}}(\mathbf{B}, \vec{b})$.

As above, fix an enumeration $\alpha_1, \dots, \alpha_N$ of the parts in diag$(\mathbf{P})$ and for each α_i, let β_i be the corresponding part in diag$(\mathbf{Q})$. Partitioning the row and column sets of S according to these 'diagonal parts', we can write the matrices S and S^{-1} in part form as

$$S = \begin{array}{c} \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{array} \!\!\begin{array}{c} \begin{array}{cccc} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{array} \\ \begin{pmatrix} S_{11} & S_{12} & \dots & S_{1N} \\ S_{21} & S_{22} & \dots & S_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N1} & S_{N2} & \dots & S_{NN} \end{pmatrix} \end{array} \text{ and } S^{-1} = \begin{array}{c} \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{array} \!\!\begin{array}{c} \begin{array}{cccc} \beta_1 & \beta_2 & \cdots & \beta_N \end{array} \\ \begin{pmatrix} T_{11} & T_{12} & \dots & T_{1N} \\ T_{21} & T_{22} & \dots & T_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ T_{N1} & T_{N2} & \dots & T_{NN} \end{pmatrix} \end{array}.$$

Observe that while S is invertible, that does not directly imply that any of the part matrices S_{ij} are themselves invertible.

Now consider some diagonal part α_i. From Lemma 4.11, the partition $\mathbf{P}$ refines the grid partition over the $\{\alpha_1, \dots, \alpha_N\}$ and also preserves atomic types. In particular, it preserves the formula

$$\varphi(x_1, x_2, \dots, x_m, y_1, y_2, \dots, y_m) := (x_1 = y_1) \wedge (x_2 = y_2) \wedge \dots \wedge (x_m = y_m)$$

which, when interpreted over α_i, defines the $\alpha_i \times \alpha_i$ identity matrix. Therefore it follows that there is some set $\mathbf{P}_i \subseteq \mathbf{P}$ of parts in $\mathbf{P}$ for which it holds that $M_i := \sum_{P \in \mathbf{P}_i} \chi_P$ is the direct sum of the identity matrix on α_i and all-zeroes matrices. That is,

$$M_i = \begin{array}{c} \\ \alpha_1 \\ \vdots \\ \alpha_{i-1} \\ \alpha_i \\ \alpha_{i+1} \\ \vdots \\ \alpha_N \end{array} \!\!\begin{array}{c} \begin{array}{ccccccc} \alpha_1 & \cdots & \alpha_{i-1} & \alpha_i & \alpha_{i+1} & \cdots & \alpha_N \end{array} \\ \begin{pmatrix} 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & I & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{pmatrix} \end{array}.$$

By Lemma 4.9, it follows that all entries of $S \cdot M_i \cdot S^{-1}$ outside the $\beta_i \times \beta_i$ sub-matrix have to be zero. Since Duplicator is playing a winning strategy (preserving atomic types), it follows that $S \cdot M_i \cdot S^{-1}$ is a sub-matrix of the $\beta_i \times \beta_i$ diagonal matrix. Furthermore, since S is invertible, we have that the diagonal matrices M_i and $S \cdot M_i \cdot S^{-1}$ have the same number of diagonal elements (for otherwise M_i and $S \cdot M_i \cdot S^{-1}$ would have different rank), which is $\|\alpha_i\| = \|\beta_i\|$. Hence, it follows that $S \cdot M_i \cdot S^{-1} = \sum_{P \in \mathbf{P}_i} f_S(\chi_P)$ is the direct sum of the identity matrix on β_i and all-zeroes matrices. Expanding the product $S \cdot M_i \cdot S^{-1}$, we have

$$S \cdot M_i \cdot S^{-1} \quad = \quad \begin{array}{c c} & \begin{array}{ccccccc} \beta_1 & \cdots & \beta_{i-1} & \beta_i & \beta_{i+1} & \cdots & \beta_N \end{array} \\ \begin{array}{c} \beta_1 \\ \vdots \\ \beta_{i-1} \\ \beta_i \\ \beta_{i+1} \\ \vdots \\ \beta_N \end{array} & \begin{pmatrix} 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & S_{ii} \cdot I \cdot T_{ii} & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{pmatrix} \end{array},$$

which shows that $S_{ii} \cdot T_{ii} = I$; that is, S_{ii} is invertible with inverse $S_{ii}^{-1} = T_{ii}$. Now by lemmas 4.11 and 4.9, we know that any $P \in \mathbf{P}$ will be within an $\alpha_i \times \alpha_j$ sub-matrix, for some i and j, and that correspondingly $f_S(P)$ will be within a $\beta_i \times \beta_j$ sub-matrix. Considering the block form of S we expressed above, it follows that the part of $f_S(P)$ falling within the $\beta_i \times \beta_j$ sub-matrix is $S_{ii} \cdot \chi_P \cdot S_{jj}^{-1}$. Therefore, writing $R_i := S_{ii}$, it can be seen that by taking $R := \bigoplus_{i=1}^{N} R_i$, we get a block-diagonal invertible matrix which, along with the partitions $\mathbf{P}$ and $\mathbf{Q}$, satisfies condition ($\circledast$) of the game and which preserves the winning strategy of the Duplicator. In particular, the induced function f_R is the same as the function f_S.

It remains to be seen that from the invertible matrix $R := \bigoplus_{i=1}^{N} R_i$, we can obtain a doubly stochastic matrix $U := \bigoplus_{i=1}^{N} U_i$ where each U_i is invertible and doubly stochastic, which (along with $\mathbf{P}$ and $\mathbf{Q}$) can also be played as a part of a winning strategy for Duplicator. To show that, consider some diagonal part α_i and let $\mathbf{P}_i \subseteq \mathbf{P}$ denote all parts in $\mathbf{P}$ lying within $\alpha_i \times \alpha_i$. Then $M_i := \sum_{P \in \mathbf{P}_i} \chi_P$ is the direct sum of all-zeroes matrices and the $\alpha_i \times \alpha_i$ matrix J_i which has 1 in every position:

$$M_i \quad = \quad \begin{array}{c c} & \begin{array}{ccccccc} \alpha_1 & \cdots & \alpha_{i-1} & \alpha_i & \alpha_{i+1} & \cdots & \alpha_N \end{array} \\ \begin{array}{c} \alpha_1 \\ \vdots \\ \alpha_{i-1} \\ \alpha_i \\ \alpha_{i+1} \\ \vdots \\ \alpha_N \end{array} & \begin{pmatrix} 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & J_i & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 0 \end{pmatrix} \end{array} \quad \text{and} \quad J_i = \begin{pmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{pmatrix}$$

Clearly, $R_i \cdot J_i \cdot R_i^{-1} =: J_i'$ is the all-ones matrix indexed by $\beta_i \times \beta_i$. That is to say, $R_i \cdot J_i = J_i' \cdot R_i$. Looking further at the matrices on each side of this equality, we see that the (x, y) entry of the $\beta_i \times \alpha_i$ matrix $R_i \cdot J_i$ is equal to the sum over all elements in the x-th row of R_i, which we denote by r_x. Similarly, the (x, y) entry of $J_i' \cdot R_i$ is equal to the sum over all elements in the y-th column of R_i, which we denote by c_y. Since $R_i \cdot J_i = J'_i \cdot R_i$, it follows that $r_x = c_y$ for all $x \in \beta_i$ and $y \in \alpha_i$; hence, $r_x = c_y = s$ for all x and y. That is to say, every row and every column of R_i sums up to s. Now observe that $s \neq 0$ (otherwise, $R_i \cdot J_i$ would have rank less than J_i, violating the assumption that R_i is invertible) and so by taking $U_i := \frac{1}{s} R_i$, we obtain the doubly stochastic and invertible matrix $U := \bigoplus_{i=1}^{N} U_i$ that was required. □

Finally, the following theorem shows that with increasing k, m or Ω, we get a decreasing chain of equivalence relations on $\text{fin}[\tau; k]$.

Theorem 4.14. For all $k, m, p \in \mathbb{N}$, with $2m \leq k$ and p prime, and all finite sets of primes Ω, it holds that $\approx^{k+1}_{m,\Omega} \subseteq \approx^{k}_{m,\Omega}$, $\approx^{k}_{m,\Omega\cup\{p\}} \subseteq \approx^{k}_{m,\Omega}$ and, if $2(m+1) \leq k$ then $\approx^{k}_{m+1,\Omega} \subseteq \approx^{k}_{m,\Omega}$.

Proof:
The first two inclusions of the lemma follow trivially from the definition of the game. To establish the last inclusion, we suppose that Duplicator has a winning strategy in the $(k, m+1, \Omega)$-invertible-map game on $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ and use that to construct for her a winning strategy in the (k, m, Ω)-game on the same game board. The new strategy is obtained by analysing at each round of the (k, m, Ω)-game the response that would be used by Duplicator in her winning strategy in the $(k, m+1, \Omega)$-game. More specifically, we simulate one move of the latter game by making the "$(k, m+1, \Omega)$-Spoiler" make the same move as he would make in the former game. Suppose that by this simulation, Duplicator will respond to Spoiler's challenge by playing $(\mathbf{P}, \mathbf{Q}, S)$ satisfying condition ($\circledast$) of the invertible-map game, where $\mathbf{P}$ and $\mathbf{Q}$ are the standard partitions of $U(\mathbf{A})^{m+1} \times U(\mathbf{A})^{m+1}$ and $U(\mathbf{B})^{m+1} \times U(\mathbf{B})^{m+1}$, respectively. Let $X_m \subseteq U(\mathbf{A})^{m+1}$ and $Y_m \subseteq U(\mathbf{B})^{m+1}$ denote the sets of all $(m+1)$-tuples of elements from $U(\mathbf{A})$ and $U(\mathbf{B})$, respectively, whose first two components are equal. Then it follows from Lemma 4.11 that every part in $\mathbf{P}$ is either a subset of X_m or contains no elements from X_m. To see this, observe that if this were not the case, then $\mathbf{P}$ would not respect equality types, which would contradict Lemma 4.11. The same argument applies to $\mathbf{Q}$ and Y_m. Therefore, we see that there are $\mathbf{P}' \subseteq \mathbf{P}$ and $\mathbf{Q}' \subseteq \mathbf{Q}$ that give partitions of X_m and Y_m, respectively. We get the desired partitions $\mathbf{P}''$ and $\mathbf{Q}''$ for Duplicator in the (k, m, Ω)-game by taking $\mathbf{P}''$ and $\mathbf{Q}''$ to be the projections of $\mathbf{P}'$ and $\mathbf{Q}'$ onto the last m components.

By Lemma 4.12, we can assume without loss of generality that the matrix S has the form $S = \bigoplus_{\alpha\in\Gamma} S_\alpha$, where Γ is the set of all atomic types on $(m+1)$-tuples realised over $U(\mathbf{A})$ and $U(\mathbf{B})$ (these are further refined by the 'diagonal types'). Let $\Gamma' \subseteq \Gamma$ denote the set of all atomic types in Γ satisfying the condition "the first and second components are equal", and let $S' := \bigoplus_{\alpha\in\Gamma'} S_\alpha$. Let S'' denote the invertible matrix obtained from S' after projecting the sets indexing the rows and columns of S' onto their last m components; that is, S'' is an $U(\mathbf{B})^m \times U(\mathbf{A})^m$ matrix while S' is an $U(\mathbf{B})^{m+1} \times U(\mathbf{A})^{m+1}$ matrix.

Now it is straightforward to verify, based on the assumption that $\mathbf{P}$, $\mathbf{Q}$ and S were played as a part of a winning strategy, that the triple $(\mathbf{P}'', \mathbf{Q}'', S'')$ satisfies condition ($\circledast$) of the invertible-map game and that any move that Spoiler makes subsequently will result in a game position from where Duplicator can also play forever in the $(k, m+1, \Omega)$-game. This inductively gives her a strategy to play forever in the (k, m, Ω)-game, as claimed. □

4.4. Composition of winning strategies

We show that the existence of a winning strategy for Duplicator in the invertible-map game is preserved under disjoint union of game boards.

First, we introduce some notation. Let A, B, C and D be sets. If $P \subseteq A \times C$ and $Q \subseteq B \times D$, then we write $P \otimes Q := \{(a, b, c, d) \mid (a, c) \in P \text{ and } (b, d) \in Q\} \subseteq A \times B \times C \times D$. Observe that $\chi_{P \otimes Q} = \chi_P \otimes \chi_Q$, where we write '$\otimes$' on the right-hand side to denote the usual matrix tensor product. Abusing notation, if $\mathbf{P}$ and $\mathbf{Q}$ are partitions of $A \times C$ and $B \times D$, respectively, then we write $\mathbf{P} \otimes \mathbf{Q} := \{P \otimes Q \mid P \in \mathbf{P} \text{ and } Q \in \mathbf{Q}\}$ to denote the *tensor partition* of $\mathbf{P}$ and $\mathbf{Q}$, which is a partition of $A \times B \times C \times D$.

Suppose $\pi \in \mathrm{Sym}(\{1, \ldots, m\})$ and $\vec{x} \in A^m$. Then we write $\pi(\vec{x}) := (x_{\pi(1)}, \ldots, x_{\pi(m)})$. If $B \subseteq A^m \times A^m$ and also $\sigma \in \mathrm{Sym}(\{1, \ldots, m\})$, then we write

$$(\pi, \sigma) \circ B := \{(\pi(\vec{x}), \sigma(\vec{y})) \mid (\vec{x}, \vec{y}) \in B\}.$$

Abusing notation, if $\mathbf{P}$ is a partition of $A^m \times A^m$, then we write $(\pi, \sigma) \circ \mathbf{P} := \{(\pi, \sigma) \circ P \mid P \in \mathbf{P}\}$.

Theorem 4.15. Suppose $(\mathbf{A}_0, \vec{a}_0) \approx^k_{m,\Omega} (\mathbf{B}_0, \vec{b}_0)$ and $(\mathbf{A}_1, \vec{a}_1) \approx^k_{m,\Omega} (\mathbf{B}_1, \vec{b}_1)$, where $\|\vec{a}_0\vec{a}_1\| = \|\vec{b}_0\vec{b}_1\| = k$. Then $(\mathbf{A}_0 \dot\cup \mathbf{A}_1, \vec{a}_0\vec{a}_1) \approx^k_{m,\Omega} (\mathbf{B}_0 \dot\cup \mathbf{B}_1, \vec{b}_0\vec{b}_1)$.

Proof:
Our aim is to show that Duplicator has a strategy to play forever in the (k, m, Ω)-game starting on $(\mathbf{A}_0 \cup \mathbf{A}_1, \vec{a}_0\vec{a}_1)$ and $(\mathbf{B}_0 \cup \mathbf{B}_1, \vec{b}_0\vec{b}_1)$.

Suppose that the game starts by Spoiler choosing a prime $p \in \Omega$ and picking up the $2m$ pairs of pebbles indexed by $\vec{j} \subseteq [k]^{2m}$. By assumption, Duplicator has winning strategies in the games played on $(\mathbf{A}_i, \vec{a}_i)$ and $(\mathbf{B}_i, \vec{b}_i)$, $i \in \{0, 1\}$. By simulating Duplicator's move for each of these games, we obtain responses $(\mathbf{P}_0, \mathbf{Q}_0, S_0)$ and $(\mathbf{P}_1, \mathbf{Q}_1, S_1)$, where the $\mathbf{P}_i$ are partitions of $A_i^m \times A_i^m$, the $\mathbf{Q}_i$ are partitions of $B_i^m \times B_i^m$, and $S_i \in \mathcal{S}_p(\mathbf{P}_i, \mathbf{Q}_i)$ for $i \in \{0, 1\}$. Write $f_i : \mathbf{P}_i \to \mathbf{Q}_i$ to denote the bijection induced by the map S_i, for $i = 0, 1$. By lemmas 4.11 and 4.12, we can obtain from these the following partitions and maps, which are fixed from now on:

- Partitions $\mathbf{P}_i^{a\times b}$ of $A_i^{a\times b}$ and $\mathbf{Q}_i^{a\times b}$ of $B_i^{a\times b}$, for all $0 \leqslant a, b \leqslant m$ and $i \in \{0, 1\}$;
- $B_i^a \times A_i^a$ invertible matrices S_i^a, for all $1 \leqslant a \leqslant m$ and $i = 0, 1$;
- bijections $f_i^{a\times b} : \mathbf{P}_i^{a\times b} \to \mathbf{Q}_i^{a\times b}$;

for which it holds that for all $1 \leqslant a, b \leqslant m$, $i \in \{0, 1\}$ and $M \in \mathbf{P}_i^{a\times b}$:

$$S_i^a \cdot \chi_M \cdot (S_i^b)^{-1} = \chi_{M'}, \tag{§}$$

where $M' := f_i^{a\times b}(M)$. We wish to use these partial partitions to construct a combined partition of the sets $(A_0 \cup A_1)^{2m}$ and $(B_0 \cup B_1)^{2m}$. This is complicated by the fact that elements of these sets can be of 'mixed type', as we explain further below. We describe the construction of the combined partition in a few steps, as follows.

1. Define a function

$$\begin{aligned} t_A : \quad & \{0,1\}^m \to \wp((A_0 \cup A_1)^m) \\ & (x_1, \ldots, x_m) \mapsto A_{x_1} \times A_{x_2} \times \cdots \times A_{x_m} \end{aligned}$$

   for partitioning the set $(A_0 \cup A_1)^m$ according to the 'signature' of the different m-tuples it contains. Define the function $t_B : \{0,1\}^m \to \wp((B_0 \cup B_1)^m)$ accordingly.

2. Enumerate $\{0,1\}^m$ in lexicographical order as $\vec{x}_1, \dots, \vec{x}_N$, for $N \geq 1$. For each $\vec{x}_i$, fix a permutation $\pi_i \in \mathrm{Sym}(\{1, \dots, m\})$ for which
$$\pi_i(\vec{x}_i) = (1^* 0^*),$$
where for a permutation π, $\pi(\vec{x}) := (x_{\pi(1)}, \dots, x_{\pi(m)})$, as defined above.

3. For each $i \in \{1, \dots, N\}$, write $\alpha_i := t_A(\vec{x}_i) \subseteq (A_0 \cup A_1)^m$ and $\beta_i := t_B(\vec{x}_i) \subseteq (B_0 \cup B_1)^m$. Clearly $\alpha_i \cap \alpha_j = \emptyset$ for $i \neq j$ and $(A_0 \cup A_1)^m = \bigcup_i \alpha_i$; similarly for the β_i's.

4. For each $i \in \{1, \dots, N\}$, suppose $\pi_i(\vec{x}_i) = (1^a 0^b)$, where $a + b = m$. Then write Π_i to denote the $\alpha_i \times (A_1^a \times A_0^b)$ permutation matrix defined by
$$\Pi_i(\vec{x}, \vec{y}) = \begin{cases} 1 & \text{if } \pi_i(\vec{x}) = \vec{y}, \\ 0 & \text{otherwise.} \end{cases}$$
We define the $\beta_i \times (B_1^a \times B_0^b)$ permutation matrix Σ_i similarly.

5. The partition of $(A_0 \cup A_1)^m$ that we consider is a certain refinement of the grid partition $\{\alpha_1, \dots, \alpha_N\} \times \{\alpha_1, \dots, \alpha_N\}$, which is illustrated in the figure below. We partition $(B_0 \cup B_1)^m$ accordingly, as will be evident later.
$$\begin{array}{c c} & \begin{array}{cccc} \alpha_1 & \alpha_2 & \cdots & \alpha_N \end{array} \\ \begin{array}{c} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{array} & \left(\begin{array}{c|c|c|c} & & \cdots & \\ \hline & & \cdots & \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline & & \cdots & \end{array}\right). \end{array}$$

6. To define this refinement, we construct separately a partition of each part $\alpha_i \times \alpha_j \subseteq (A_0 \cup A_1)^m \times (A_0 \cup A_1)^m$. Suppose $\pi_i(\vec{x}_i) = (1^a 0^b)$ and $\pi_j(\vec{x}_j) = (1^c 0^d)$, where $0 \leqslant a, b, c, d \leqslant m$ with $a + b = c + d = m$. As noted before, we have partitions $\mathbf{P}_1^{a \times c}$ and $\mathbf{P}_0^{b \times d}$ of $A_1^a \times A_1^c$ and $A_0^b \times A_0^d$, respectively. Therefore, their tensor $\mathbf{P}_1^{a \times c} \otimes \mathbf{P}_0^{b \times d}$ is a partition of $A_1^a \times A_0^b \times A_1^c \times A_0^d \subseteq (A_0 \cup A_1)^{2m}$. We obtain the desired partition of $\alpha_i \times \alpha_j$ by taking
$$R_{ij} := (\pi_i^{-1}, \pi_j^{-1}) \circ (\mathbf{P}_1^{a \times c} \otimes \mathbf{P}_0^{b \times d}).$$

7. Finally, we get a partition $\mathbf{P}$ of $(A_0 \cup A_1)^{2m}$ by taking $\mathbf{P} := \{R_{ij} \mid 1 \leqslant i, j \leqslant N\}$. We repeat the same process to get a partition $\mathbf{Q}$ of $(B_0 \cup B_1)^{2m}$, by constructing a tensor partition of each part $\beta_i \times \beta_j$ just like we did above.

It remains to be shown how the partitions $\mathbf{P}$ and $\mathbf{Q}$ can be related by a single invertible map. So consider $i \leqslant N$ and suppose $\pi_i(\vec{x}_i) = (1^a 0^b)$, where $a + b = m$. Let S_1^a and S_0^b be two of the invertible matrices we introduced before, where S_1^a is indexed by $B_1^a \times A_1^a$ and S_0^b is indexed by $B_0^b \times A_0^b$. Let $S_{i,1} := S_1^a$ and $S_{i,0} := S_0^b$ and define
$$T_i := \Sigma_i \cdot (S_{i,1} \otimes S_{i,0}) \cdot \Pi_i^{-1}.$$

Here, $S_{i,1} \otimes S_{i,0}$ is a $(B_1^a \times B_0^b) \times (A_1^a \times A_0^b)$ matrix and Σ_i, Π_i are permutation matrices indexed by $\beta_i \times (B_1^a \times B_0^b)$ and $\alpha_i \times (A_1^a \times A_0^b)$, respectively. Thus, T_i is indexed by $\beta_i \times \alpha_i$. Finally, we let

$$T := \bigoplus_{i=1}^{N} T_i.$$

Whilst T is clearly invertible, since each of its diagonal parts is invertible, it remains to be shown that T preserves winning positions in each of the sub-games played on $(\mathbf{A}_i, \vec{a}_i)$ and $(\mathbf{B}_i, \vec{b}_i)$, $i \in \{0,1\}$. So consider some part $P \in \alpha_i \times \alpha_j$. It follows from our construction that each such P can be written uniquely as $P = (\pi_i^{-1}, \pi_j^{-1}) \circ (P_1 \otimes P_0)$, where $P_0 \in \mathbf{P}_0$ and $P_1 \in \mathbf{P}_1$. Writing M, M_0 and M_1 for the characteristic matrices of P, P_0 and P_1, respectively, this implies that the $\alpha_i \times \alpha_j$ matrix M can be written as

$$M = \Pi_i \cdot (M_1 \otimes M_0) \cdot \Pi_j^{-1},$$

where '$\otimes$' here denotes the usual matrix tensor product. The next claim shows that the map T correctly preserves winning positions in the two sub-games, which concludes the proof of the theorem.

Claim 4.16. Let $M_i' = \chi_{f_i(P_i)}$ for $i \in \{0,1\}$. Then

$$\begin{aligned} T_i \cdot M \cdot T_j^{-1} &= T_i \cdot (\Pi_i \cdot (M_1 \otimes M_0) \cdot \Pi_j^{-1}) \cdot T_j^{-1} \\ &= \Sigma_i \cdot (M_1' \otimes M_0') \cdot \Sigma_j^{-1}. \end{aligned}$$

Proof of claim.

$$\begin{aligned} T_i \cdot M \cdot T_j^{-1} &= (\Sigma_i \cdot (S_{i,1} \otimes S_{i,0}) \cdot \Pi_i^{-1}) \cdot (\Pi_i \cdot (M_1 \otimes M_0) \cdot \Pi_j^{-1}) \cdot (\Sigma_j \cdot (S_{j,1} \otimes S_{j,0}) \cdot \Pi_j^{-1})^{-1} \\ &= (\Sigma_i \cdot (S_{i,1} \otimes S_{i,0}) \cdot \Pi_i^{-1}) \cdot (\Pi_i \cdot (M_1 \otimes M_0) \cdot \Pi_j^{-1}) \cdot (\Pi_j \cdot (S_{j,1} \otimes S_{j,0})^{-1} \cdot \Sigma_j^{-1}) \\ &= \Sigma_i \cdot (S_{i,1} \otimes S_{i,0}) \cdot (M_1 \otimes M_0) \cdot (S_{j,1}^{-1} \otimes S_{j,0}^{-1}) \cdot \Sigma_j^{-1} \\ &= \Sigma_i \cdot (S_{i,1} \cdot M_1 \cdot S_{j,1}^{-1}) \otimes (S_{i,0} \cdot M_0 \cdot S_{j,0}^{-1}) \cdot \Sigma_j^{-1} \\ &= \Sigma_i \cdot (M_1' \otimes M_0') \cdot \Sigma_j^{-1}. \end{aligned}$$

□

4.5. Complexity of the game equivalence

In this section we show that for each k and vocabulary τ, there is an algorithm that decides whether $(\mathbf{A}, \vec{a}) \approx_{m,\Omega}^k (\mathbf{B}, \vec{b})$ in time polynomial in $n \cdot p_{\max}$, where n is the size of both $\mathbf{A}$ and $\mathbf{B}$ and $p_{\max}$ is the largest prime in Ω.

Consider some $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ in $\text{fin}[\tau; k]$ and assume that $\|\mathbf{A}\| = \|\mathbf{B}\| = n$. Since the number of distinct positions in the game starting with $(\mathbf{A}, \vec{a})$ and $(\mathbf{B}, \vec{b})$ is bounded by a polynomial in n, it follows that there is some polynomial $q : \mathbb{N} \to \mathbb{N}$ (depending only on k and τ) such that if Duplicator can play the game for at least $q(n)$ rounds, then she has a strategy to play forever. In other words, $(\mathbf{A}, \vec{a}) \approx_{m,\Omega}^k (\mathbf{B}, \vec{b})$ if, and only if, $(\mathbf{A}, \vec{a}) \sim_{q(n)} (\mathbf{B}, \vec{b})$. To decide $(\mathbf{A}, \vec{a}) \approx_{m,\Omega}^k (\mathbf{B}, \vec{b})$, we inductively construct the graph of $\approx_{m,\Omega}^k$, restricted to $\mathbf{A}$ and $\mathbf{B}$, as follows. Initially, we partition the elements of $U(\mathbf{A})^k \mathbin{\dot\cup} U(\mathbf{B})^k$ by their atomic equivalance, which is just $\sim_0$. For the induction step, suppose we have constructed $\sim_i$. Then to compute the refinement $\sim_{i+1}$, we consider each $\sim_i$-equivalent pair $(\vec{c}, \vec{d})$

and check whether condition ($\star\star$) of Lemma 4.2 is satisfied. That is, for each $p \in \Omega$ and $\vec{j} \in [k]^{2m}$, we let $\mathcal{C} = (C_\alpha)$ and $\mathcal{D} = (D_\alpha)$ be the families of extension matrices defined by $\vec{j}$ over $\vec{c}$ and $\vec{d}$, respectively, indexed by all equivalence classes α of $\sim_i$ (where $C_\alpha = \mathrm{extmat}^{\alpha}_{\vec{j}}(\mathbf{A}, \vec{c})$ if $\vec{c}$ is defined over $\mathbf{A}$, and similarly for D_α). Here it is important to note that it suffices to consider only equivalence classes of $\sim_i$ *restricted to* $\mathbf{A}$ *and* $\mathbf{B}$. Therefore, the number of extension matrices that we need to consider is bounded by a polynomial in n.

At this stage it remains to determine whether the pair of matrix tuples $\mathcal{C}$ and $\mathcal{D}$ are *simultaneously similar*: that is, whether there is a non-singular matrix S such that $S \cdot C_\alpha \cdot S^{-1} = D_\alpha$ for all $C_\alpha \in \mathcal{C}$. By a result of Chistov et al. [9], this problem is in polynomial time over all finite fields.

Proposition 4.17. (Chistov, Karpinsky and Ivanyov)
There is a deterministic algorithm that, given two families of $N \times N$ matrices $\mathcal{C} = (C_1, \ldots, C_l)$ and $\mathcal{D} = (D_1, \ldots, D_l)$ over a finite field GF_q, determines in time $\mathrm{poly}(N, l, q)$ whether $\mathcal{C}$ and $\mathcal{D}$ are simultaneously similar.

By our discussion above, it follows that we can construct the graph of $\approx^k_{m,\Omega}$ restricted to $\mathbf{A}$ and $\mathbf{B}$ in a polynomial number of steps. At each step, we need to check a polynomial number of matrix tuples for simultaneous similarity, where each tuple has polynomial length. This gives us a proof of the following theorem.

Theorem 4.18. For each τ there is a deterministic algorithm that, given $(\mathbf{A}, \vec{a}), (\mathbf{B}, \vec{b}) \in \mathrm{fin}[\tau; k]$ (with $\|\mathbf{A}\| = \|\mathbf{B}\| = n$), $m \in \mathbb{N}$ with $2m \leq k$ and a finite set of primes Ω, decides whether $(\mathbf{A}, \vec{a}) \approx^k_{m,\Omega} (\mathbf{B}, \vec{b})$ in time $(np)^{\mathcal{O}(k)}$ where p is the largest prime in Ω.

Observe that this implies that for each fixed k, we can decide $\approx^k_{m,\Omega}$ in polynomial time, where Ω can be a part of the input and $m \leq k$.

5. Application to the graph isomorphism problem

By considering the invertible-map game equivalence $\approx^k_{m,\Omega}$ on the class of all finite graphs, we get a family of polynomial-time algorithms for stratifying the graph isomorphism relation. More specifically, for each k, m and Ω, we write $\mathrm{IM}^k_{m,\Omega}$ to denote the following algorithm on a pair of finite graphs $\mathbf{G}$ and $\mathbf{H}$:

> If $\|\mathbf{G}\| \neq \|\mathbf{H}\|$ then output "not isomorphic". Otherwise, compute $\approx^k_{m,\Omega}$ (restricted to $\mathbf{G}$ and $\mathbf{H}$) on the set $U(\mathbf{G})^k \,\dot{\cup}\, U(\mathbf{H})^k$ by applying the algorithm of Theorem 4.18 for all tuples in $U(\mathbf{G})^k \,\dot{\cup}\, U(\mathbf{H})^k$. If the result is that there is some equivalence class α of $\approx^k_{m,\Omega}$ such that $\|\alpha \cap U(\mathbf{G})^k\| \neq \|\alpha \cap U(\mathbf{H})^k\|$ then output "not isomorphic"; else output "isomorphic".

It follows from Theorem 4.18 that $\mathrm{IM}^k_{m,\Omega}$ runs in polynomial time for a fixed k. While the algorithm will always correctly identify isomorphic graphs, it may fail to distinguish between non-isomorphic instances. Furthermore, it can be seen that for each pair of graphs, there is always a value of k for which $\mathrm{IM}^k_{m,\Omega}$ correctly determines isomorphism for any m and any finite set of primes Ω. In particular, this is so when k is at least as large as the number of vertices in each of the graphs.

The procedure for $\mathrm{IM}^k_{m,\Omega}$ that we outlined above bears a strong resemblance to the well-known Weisfeiler-Leman method for graph isomorphism (see [4] for a description of the method). It was shown by Cai, Fürer and Immerman [4] that Duplicator has a winning strategy in the $(k+1)$-pebble bijection game on $\mathbf{G}$ and $\mathbf{H}$ if and only if $\mathbf{G}$ and $\mathbf{H}$ are not distinguished by the k-dimensional Weisfeiler-Leman algorithm (WL^k). Combining this characterisation of the Weisfeiler-Leman algorithm with lemmas 3.4 and 4.1, we have that

$$\begin{aligned} &\mathbf{G} \text{ and } \mathbf{H} \text{ are distinguished by } \mathrm{WL}^k \\ \Rightarrow\ &\mathbf{G} \text{ and } \mathbf{H} \text{ are distinguished by } \mathrm{IM}^{k+2m}_{m,\Omega} \text{ for } \textit{all } m \text{ and prime sets } \Omega. \end{aligned}$$

In [4], Cai et al. showed how to construct for each $k \in \mathbb{N}$ a pair of non-isomorphic graphs (named "CFI graphs") that are indistinguishable in WL^k. Later, it was shown by Dawar et al. [5] that there is a fixed sentence of first-order logic with rank operators over GF_2 that can distinguish between any pair of these CFI graphs. This construction was extended by Holm [12], who showed that for any prime p, there are families of non-isomorphic graphs that can be distinguished by first-order logic with rank operators over GF_p but not by any fixed dimension of Weisfeiler-Leman. Hence, it follows that the family of $\mathrm{IM}^k_{m,\Omega}$ algorithms provide a way of stratifying the graph isomorphism relation which goes beyond that given by the Weisfeiler-Leman algorithms, and we get the following proposition.

Proposition 5.1. For each prime p and $k \geq 1$, there is a pair of non-isomorphic graphs $\mathbf{G}$ and $\mathbf{H}$ that can be distinguished by $\mathrm{IM}^3_{\{p\},1}$ but not by WL^k.

Finally, we remark that Derksen [20] has recently described a family of polynomial-time algorithms that also give an approximation to graph isomorphism that goes beyond that of the Weisfeiler-Leman method. While Derksen's method partly builds on the simultaneous-similarity algorithm of Chistov et al. [9] (Proposition 4.17), it also draws heavily on techniques from algebraic geometry and category theory and seems very different from the game-based approach that we describe. Nevertheless, it is an interesting open problem whether these two approaches can be related.

6. Discussion

A natural question that is raised by the definitions of the games we have presented in this paper, is how to use them to establish inexpressibility results. A step in this direction is presented in [12] where it is shown that for any prime p, there is a property definable in first-order logic with rank operators over GF_p which is not closed under $\equiv^R_{k,1,\{q\}}$ for any k and primes $q \neq p$. This method can be further extended to work for all sets of primes $\Omega \neq \Gamma$ rather than just single primes. It remains a challenge to lift this up to arities higher than 1, since playing the game poses combinatorial difficulties. Grädel and Pakusa [6] offer an alternative approach, not based on games, of showing that the rank of matrices with respect to GF_p is not definable in the logic IFPR using only rank operators $\mathbf{rk}^q$ for $q \neq p$. It seems plausible that their construction yields examples of non-isomorphic structures that are not distinguished by $\equiv^R_{k,m,\Omega}$ for any k, m and Ω, but this remains to be verified.

Another interesting direction would be to establish the precise relationship between the two games we consider. While we showed that the invertible-map game gives a refinement of the matrix-equivalence game (that is, a winning strategy for Duplicator in the former gives a winning strategy in the latter), it is not known whether this refinement is strict. Might it be the case that for any k and m one can find a k' and m' so that $\equiv^R_{k',m',\Omega}$ is a refinement of $\approx^k_{m,\Omega}$? One way this might be established is by showing that the relations $\approx^k_{m,\Omega}$ are themselves definable in IFPR. If it turns out that this is not the case, then we would have established that there is a PTIME property not in in IFPR. A natural line of investigation would then be to extract from the invertible-map game a suitable logical operator, stronger than the matrix-rank operator, that is characterised by this game.

A more general direction of research that is suggested by this work is to explore other partition games which can be defined by suitable equivalence conditions on the partition matrices. There is space here for defining new logics and also new isomorphism tests.

Finally, there is also scope to extend the partition games that we consider so that they can be applied to express non-definability results for the logic FPR^* used by Grädel and Pakusa [6]. Recall that in this logic, the prime over which the rank is computed is not fixed but included as a part of the input. In [6] it is shown that the logic FPR^* strictly extends the expressive power of IFPR while still being contained in PTIME, and it remains open whether it captures all of polynomial time. One way of extending the matrix-equivalence game is to allow the game players to consider all primes $p \leq n$, where n is the maximum cardinality of the two finite game structures, instead of playing with a fixed set of primes Ω. It can be shown that this extension of the game provides a game method for proving inexpressibility results in the logic FPR^*.

Acknowledgements. This research was partially done whilst the authors were participants in the programme *Semantics and Syntax* at the Isaac Newton Institute for the Mathematical Sciences in Cambridge, UK.

References

[1] Gurevich Y. Logic and the Challenge of Computer Science. In: Börger E, editor. Current Trends in Theoretical Computer Science. Computer Science Press; 1988. p. 1–57.

[2] Immerman N. Relational Queries computable in Polynomial Time. Information and Control. 1986;68:86–104. doi:10.1016/s0019-9958(86)80029-8.

[3] Vardi MY. The Complexity of Relational Query Languages. In: STOC '82: Proc. 14th ACM Symp. on Theory of Computing. ACM Press; 1982. p. 137–146. doi:10.1145/800070.802186.

[4] Cai JY, Fürer M, Immerman N. An Optimal Lower Bound on the Number of Variables for Graph Identification. Combinatorica. 1992;12(4):389–410. doi:10.1007/bf01305232.

[5] Dawar A, Grohe M, Holm B, Laubner B. Logics with Rank Operators. In: Proc. 24th IEEE Symp. on Logic in Computer Science; 2009. p. 113–122. doi:10.1109/lics.2009.24.

[6] Grädel E, Pakusa W. Rank Logic is Dead, Long Live Rank Logic! In: 24th EACSL Annual Conference on Computer Science Logic, CSL 2015; 2015. p. 390–404. doi:10.4230/LIPIcs.CSL.2015.390.

[7] Immerman N, Lander E. Describing graphs: A first-order approach to graph canonization. Complexity Theory Retrospective: In Honor of Juris Hartmanis on the Occasion of His Sixtieth Birthday, July 5, 1988. 1990;p. 59—81. doi:10.1007/978-1-4612-4478-3_5.

[8] Hella L. Logical hierarchies in PTIME. Information and Computation. 1996;129:1—19. doi:10.1006/inco.1996.0070.

[9] Chistov A, Ivanyos G, Karpinski M. Polynomial time algorithms for modules over finite dimensional algebras. In: Proc. 1997 international symposium on symbolic and algebraic computation. ACM; 1997. p. 68–74. doi:10.1145/258726.258751.

[10] Ebbinghaus HD, Flum J. Finite Model Theory. Springer-Verlag; 1999. doi:10.1007/3-540-28788-4.

[11] Libkin L. Elements of finite model theory. Springer-Verlag; 2004. doi:10.1007/978-3-662-07003-1.

[12] Holm B. Descriptive complexity of linear algebra. Univ. of Cambridge; 2010.

[13] Otto M. Bounded Variable Logics and Counting — A Study in Finite Models. vol. 9 of Lecture Notes in Logic. Springer-Verlag; 1997. Available from: `http://projecteuclid.org/euclid.lnl/1235416490`.

[14] Laubner B. The Structure of Graphs and New Logics for the Characterization of Polynomial Time. Humboldt-Universität zu Berlin; 2011.

[15] Fraïssé R. Sur quelques Classifications des Systèmes de Relations. Publications Scientifiques de l'Université d'Algerie, Séries A. 1954;1:35–182.

[16] Ehrenfeucht A. An application of games to the completeness problem for formalized theories. Fundamenta Mathematicae. 1961;49(129-141):13. Available from: `http://pldml.icm.edu.pl/pldml/element/bwmeta1.element.bwnjournal-article-fmv49i2p129bwm`.

[17] Barwise J. On Moschovakis Closure Ordinals. Journal of Symbolic Logic. 1977;42:292–296. doi:10.2307/2272133.

[18] Poizat B. Deux Ou Trois Choses Que je Sais de L_n. Journal of Symbolic Logic. 1982;47(3):641–658. doi:10.2307/2273594.

[19] Dawar A, Holm B. Pebble Games for Logics with Counting and Rank. In: Cégielski P, editor. Studies in Weak Arithmetics. CSLI Publications; 2010. p. 99–120.

[20] Derksen H. The Graph Isomorphism Problem and approximate categories. J Symb Comput. 2013;59:81–112. doi:10.1016/j.jsc.2013.06.002.